



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**  
**CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y REDES**  
**INDUSTRIALES**

**“DISEÑO E IMPLEMENTACIÓN DE UNA INTERFAZ PARA LA**  
**TRADUCCIÓN DE CARACTERES DEL ALFABETO**  
**DACTILOLÓGICO A PALABRAS UTILIZANDO SENSORES**  
**MUSCULARES Y RECONOCIMIENTO DE PATRONES”**

**TRABAJO DE TITULACIÓN**

**Tipo:** Dispositivo Tecnológico

**Presentado para optar al Grado Académico de:**

**INGENIERA EN ELECTRÓNICA, CONTROL Y REDES**  
**INDUSTRIALES**

**AUTORA:** Ñacato Estrella Ana María

**DIRECTOR:** Ing. Jorge Hernández Ambato

Riobamba – Ecuador  
2019

**©2019, Ñacato Estrella Ana María**

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**  
**CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y REDES INDUSTRIALES**

El Tribunal de Tesis certifica que: El trabajo técnico: “DISEÑO E IMPLEMENTACIÓN DE UNA INTERFAZ PARA LA TRADUCCIÓN DE CARACTERES DEL ALFABETO DACTILOLÓGICO A PALABRAS UTILIZANDO SENSORES MUSCULARES Y RECONOCIMIENTO DE PATRONES.”, de responsabilidad de la señorita Ñacato Estrella Ana María, ha sido minuciosamente revisado por los Miembros del Tribunal de Tesis, quedando autorizada su presentación.

<b>NOMBRE</b>	<b>FIRMA</b>	<b>FECHA</b>
Dr. Washington Luna E. <b>DECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA</b>	_____	_____
Ing. Freddy Chávez V. <b>DIRECTOR DE ESCUELA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y REDES INDUSTRIALES</b>	_____	_____
Ing. Jorge Hernández A. <b>DIRECTOR DEL TRABAJO DE TITULACIÓN</b>	_____	_____
Ing. Edwin Altamirano S. <b>MIEMBRO DEL TRABAJO DE TITULACIÓN</b>	_____	_____

Yo, Ana María Ñacato Estrella, declaro ser responsable de las ideas, doctrinas y resultados expuestos en este Trabajo de Titulación y el patrimonio intelectual le pertenece a la Escuela Superior Politécnica de Chimborazo.

.....  
Ana María Ñacato Estrella

**ESTUDIANTE**

## **DEDICATORIA**

A mis padres y hermano por ser un ejemplo de lucha y constancia diaria para alcanzar y finalizar las metas planteadas.

A mi familia por todo el apoyo incondicional brindado durante esta etapa universitaria.

*Ana María*

## **AGRADECIMIENTO**

A Dios Todopoderoso, que en su infinita bondad iluminó mi camino y me fortaleció día a día.

A mis padres y hermano, por todo el amor, cariño y apoyo que recibo diariamente.

A los docentes de las Escuela Superior Politécnica de Chimborazo, por todo el esfuerzo y dedicación que realizan al cumplir una labor tan noble como la de enseñar, de manera especial al Ing. Jorge Hernández e Ing. Edwin Altamirano por el apoyo y consejos brindados para la culminación de este trabajo de titulación.

*Ana María*

## TABLA DE CONTENIDO

RESUMEN.....	xvi
ABSTRACT .....	xvii
INTRODUCCIÓN .....	1
CAPÍTULO I.....	6
1. MARCO TEÓRICO REFERENCIAL .....	6
1.1. Discapacidad Auditiva.....	6
1.1.1. Métodos oralistas .....	6
1.1.2. Métodos gestualistas .....	7
1.1.3. Métodos Mixtos .....	7
1.2. Alfabeto dactilológico .....	7
1.3. Electromiografía (EMG).....	8
1.3.1. Electromiografía invasiva.....	9
1.3.2. Electromiografía de superficie (SEMG) o no invasiva .....	10
1.3.3. Bases fisiológicas .....	11
1.4. Sensores .....	13
1.4.1. Sensores musculares superficiales.....	13
1.4.1.1. Sensor muscular MyoWare.....	13
1.4.1.2. Sensor muscular EMG (Electromiografía).....	14
1.4.1.3. Myo armband.....	15
1.4.2. Análisis comparativos de sensores musculares .....	16
1.5. Sistema de posicionamiento de electrodos.....	17
1.6. Procesamiento de señales EMG.....	18
1.6.1. Transformada de Fourier.....	18
1.6.2. Transformada de Fourier de Tiempo Corto .....	19
1.6.3. Transformada Wavelet .....	21
1.6.3.1. Transformada Wavelet Continua CWT.....	22
1.6.3.2. Transformada Wavelet Discreta DWT.....	23

1.6.4.	<i>Pywavelet</i> .....	27
1.7.	<b>Aprendizaje Supervisado</b> .....	28
1.7.1.	<i>Modelos de Clasificación</i> .....	28
1.7.1.1.	<i>Árbol de decisión</i> .....	29
1.7.1.2.	<i>Random Forest</i> .....	30
1.7.1.3.	<i>Redes Neuronales Artificiales</i> .....	30
1.7.2.	<i>Evaluación de modelos</i> .....	32
1.7.2.1.	<i>Matriz de confusión</i> .....	33
1.8.	<b>Sistemas Embebidos</b> .....	35
1.8.1.	<i>Plataforma embebida – FPGA</i> .....	36
1.8.1.1.	<i>Digilent PYNQ-Z</i> .....	36
<b>CAPÍTULO II</b> .....		39
2.	<b>MARCO METODOLÓGICO</b> .....	39
2.1.	<b>Adquisición de señales EMG</b> .....	39
2.1.1.	<i>Ubicación de sensor</i> .....	40
2.1.2.	<i>Software de adquisición</i> .....	40
2.2.	<b>Sistema de procesamiento de señales</b> .....	42
2.2.1.	<i>Aplicación de la Transformada Wavelet – Análisis Multirresolución</i> .....	42
2.2.1.1.	<i>Reconstrucción de la señal</i> .....	44
2.2.2.	<i>Extracción de características</i> .....	44
2.2.2.1.	<i>Valor Absoluto Medio (MAV)</i> .....	45
2.2.2.2.	<i>Root Mean Squared (RMS)</i> .....	46
2.2.2.3.	<i>Waveform Length (WL)</i> .....	46
2.3.	<b>Aprendizaje automático con modelos de Clasificación</b> .....	46
2.3.1.	<i>Preparación del conjunto de datos</i> .....	46
2.3.2.	<i>Modelos de aprendizaje</i> .....	49
2.3.2.1.	<i>Análisis de los datos obtenidos</i> .....	50
2.3.2.2.	<i>Red Neuronal Multi-Layer Perceptron</i> .....	51
2.3.2.3.	<i>Árbol de Decisión</i> .....	52



2.3.2.4.	<i>Random Forest</i> .....	53
2.4.	Diseño de la interfaz .....	53
2.5.	Implementación sobre la FPGA .....	54
2.5.1.	<i>Diagrama de bloques de la interfaz de traducción del alfabeto dactilológico</i> .....	55
2.5.2.	<i>Diagrama de flujo del funcionamiento de la interfaz de traducción del alfabeto dactilológico</i> .....	56
CAPÍTULO III.....		57
3.	MARCO DE RESULTADOS .....	57
3.1.	Reconstrucción de señales .....	58
3.2.	Clasificación de señas .....	60
3.2.1.	<i>Elección de parámetros</i> .....	60
3.2.1.1.	<i>Multi Layer Perceptron</i> .....	60
3.2.1.2.	<i>Árbol de decisión</i> .....	63
3.2.1.3.	<i>Random Forest</i> .....	64
3.2.2.	<i>Pruebas de datos correctamente clasificados</i> .....	66
CONCLUSIONES.....		71
RECOMENDACIONES.....		72
BIBLIOGRAFÍA		
ANEXOS		

## INDICE DE TABLAS

<b>Tabla 1-1:</b>	Comparación de características entre sensores musculares superficiales.....	16
<b>Tabla 2-1:</b>	Matriz de confusión para una clasificación binaria .....	33
<b>Tabla 3-1:</b>	Características FPGA PYNQ-Z1 .....	37
<b>Tabla 1-2:</b>	Evaluación del conjunto de datos con varios clasificadores.....	50
<b>Tabla 2-2:</b>	Configuración de parámetros para <code>sk.neural_network.MLPClassifier</code> .....	51
<b>Tabla 3-2:</b>	Configuración de parámetros para <code>sk.DecisionTreeClassifier</code> .....	52
<b>Tabla 4-2:</b>	Parámetros para <code>sk.RandomForestClassifier</code> .....	53
<b>Tabla 1-3:</b>	Datos de prueba .....	57
<b>Tabla 2-3:</b>	Promedio: Error cuadrático medio para cada una de las letras.....	59
<b>Tabla 3-3:</b>	Porcentaje de clasificación correctamente para varios modelos MLP .....	61
<b>Tabla 4-3:</b>	Porcentaje de clasificación para el modelo Árbol de decisión, variando la profundidad máxima .....	63
<b>Tabla 5-3:</b>	Porcentaje de datos bien clasificados para el modelo Random Forest, variando el número de árboles.....	64
<b>Tabla 6-3:</b>	Parámetros de los modelos y su porcentaje de clasificación .....	66

## ÍNDICE DE FIGURAS

<b>Figura 1-1:</b>	Alfabeto dactilológico ecuatoriano .....	8
<b>Figura 2-1:</b>	Electromiografía invasiva .....	9
<b>Figura 3-1:</b>	Electromiografía superficial, señal EMG.....	10
<b>Figura 4-1:</b>	Espectro de frecuencias de varias señales EMG .....	10
<b>Figura 5-1:</b>	Componentes de la unidad motora.....	12
<b>Figura 6-1:</b>	Representación esquemática de los mecanismos básicos de control .....	13
<b>Figura 7-1:</b>	Sensor muscular MyoWare con electrodos.....	14
<b>Figura 8-1:</b>	Sensor muscular EMG .....	14
<b>Figura 9-1:</b>	Myo Armband A. Vista externa B. Elementos que conforman el módulo principal	15
<b>Figura 10-1:</b>	Ocho electrodos de superficie EMG en Myo Armband.....	15
<b>Figura 11-1:</b>	Raw data de cada sensor EMG .....	16
<b>Figura 12-1:</b>	Ubicación de electrodos superficiales en brazos y manos. A. Vista frontal, B. Vista Posterior .....	17
<b>Figura 13-1:</b>	Transformación del dominio del tiempo al dominio de la frecuencia de una señal.	18
<b>Figura 14-1:</b>	Señales con su espectro de frecuencias para Señal estacionaria vs señal no-estacionaria .....	19
<b>Figura 15-1:</b>	SFTF en una señal $x(t)$ .....	20
<b>Figura 16-1:</b>	Tamaño de la ventana y su influencia en la resolución tiempo-frecuencia.....	20
<b>Figura 17-1:</b>	Se observa algunas de las principales familias wavelets .....	22
<b>Figura 18-1:</b>	Aplicación de la CWT a una señal.....	23
<b>Figura 19-1:</b>	Diagrama esquemático de la descomposición de la señal original (S) en una señal de alta frecuencia (coeficientes de detalle) y una señal de baja frecuencia (Coeficientes de aproximación).....	25
<b>Figura 20-1:</b>	Descomposición wavelet de nivel 5, a una señal con frecuencia de 1000Hz, de color rojo tenemos los coeficientes de aproximación y de color verde los coeficientes de detalle. ....	26
<b>Figura 21-1:</b>	Coeficientes de reconstrucción .....	27
<b>Figura 22-1:</b>	Aprendizaje supervisado.....	28
<b>Figura 23-1:</b>	Arboles de decisión.....	29
<b>Figura 24-1:</b>	Red neuronal biológica vs red neuronal artificial .....	30
<b>Figura 25-1:</b>	Red neuronal Multi capa.....	31

<b>Figura 26-1:</b>	Función sigmoid.....	32
<b>Figura 27-1:</b>	Framework de código abierto PYNQ.....	37
<b>Figura 28-1:</b>	Jupyter Notebook para PYNQ-Z1.....	38
<b>Figura 1-2:</b>	Diagramas de bloques del sistema .....	39
<b>Figura 2-2:</b>	Sensor Myo, A. Antebrazo derecho, B. Músculos del antebrazo y Myo Armband .	40
<b>Figura 3-2:</b>	Interfaz de adquisición de señales EMG.....	41
<b>Figura 4-2:</b>	Archivo CSV con datos adquiridos mediante el Myo armband.....	42
<b>Figura 5-2:</b>	DWT nivel de descomposición 2 .....	43
<b>Figura 6-2:</b>	Señal S en azul, A <sub>1</sub> , A <sub>2</sub> en rojo, D <sub>1</sub> , D <sub>2</sub> color verde .....	43
<b>Figura 7-2:</b>	Reconstrucción de la señal original desde un nivel de descomposición 2 .....	44
<b>Figura 8-2:</b>	Procesamiento y extracción de características .....	45
<b>Figura 9-2:</b>	Matriz del conjunto de datos.....	47
<b>Figura 10-2:</b>	Visualización del conjunto de datos. A. Primeras 5 filas, B. Últimas 5 filas.....	47
<b>Figura 11-2:</b>	Información del conjunto de datos.....	48
<b>Figura 12-2:</b>	Distribución de datos .....	48
<b>Figura 13-2:</b>	Procedimiento para la clasificación de datos .....	49
<b>Figura 14-2:</b>	División de los datos.....	49
<b>Figura 15-2:</b>	Interfaz de traducción .....	54
<b>Figura 16-2:</b>	Librerías utilizadas a través de Jupyter Notebook de la FPGA .....	55
<b>Figura 17-2:</b>	Diagrama de bloques de la interfaz de traducción del alfabeto dactilológico.....	55
<b>Figura 18-2:</b>	Diagrama de flujo, A. Interfaz de traducción, B. Procesamiento de la FPGA.....	56
<b>Figura 1-3:</b>	Matriz de confusión del modelo MLP .....	67
<b>Figura 2-3:</b>	Matriz de confusión del modelo Árbol de decisión .....	68
<b>Figura 3-3:</b>	Matriz de confusión del modelo Random Forest .....	69

## ÍNDICE DE GRÁFICOS

<b>Gráfico 1-3:</b>	Error cuadrático medio del electrodo 1 para cada una de las letras.....	60
<b>Gráfico 2-3:</b>	Gráfica del conjunto de datos evaluados con 2, 3 y 4 capas ocultas .....	62
<b>Gráfico 3-3:</b>	Relación entre la variación de la profundidad máxima del árbol vs porcentaje de clasificación.....	64
<b>Gráfico 4-3:</b>	Relación entre la variación del número de árboles del bosque vs porcentaje de clasificación.....	66
<b>Gráfico 5-3:</b>	Relación del número de datos verdaderos por cada clase vs datos correctamente clasificados por cada modelo. ....	70

## ÍNDICE DE ANEXOS

**Anexo A:** Error cuadrático medio

**Anexo B:** Número de datos verdaderos por cada clase vs número de datos clasificados correctamente por cada modelo

**Anexo C:** Métrica de evaluación: precisión y recall de cada modelo

## ÍNDICE DE ABREVIATURAS

<b>CONADIS:</b>	Consejo Nacional para la Igualdad de Discapacidades
<b>RNA:</b>	Redes neuronales artificiales
<b>IMEn:</b>	Intrinsic-mode entropy
<b>GSL:</b>	Lengua de signos griegos
<b>ALVOT:</b>	Algoritmo de votación
<b>ESPOCH:</b>	Escuela Superior Politécnica de Chimborazo
<b>LSEC:</b>	Lengua de señas ecuatoriana
<b>UM:</b>	Unidad motora
<b>μV:</b>	Micro voltios
<b>mV:</b>	Mili voltios
<b>EMG:</b>	Electromiografía
<b>CNE:</b>	Concentric needle electrode (electrodo de aguja concéntrico)
<b>SFEMG:</b>	Single-Fiber EMG (electromiografía de fibra única)
<b>SEMG:</b>	Electromiografía de superficie o superficial
<b>Vs:</b>	Voltaje de la fuente de energía

## RESUMEN

En el presente trabajo de titulación se desarrolló un sistema dirigido a personas con discapacidad auditiva que hacen uso del alfabeto dactilológico para su comunicación. Este dispositivo permite la traducción de los símbolos del alfabeto convencional representadas en señas a letras utilizando modelos de aprendizaje automático supervisado. La implementación, procesamiento y clasificación se ejecuta sobre una plataforma embebida FPGA PYNQ-Z1, la misma que es desarrollada bajo el sistema de código abierto Linux compatible con el programa Python y sus diferentes librerías. El proceso se desarrolla inicialmente con la adquisición de los datos por medio del sensor muscular Myo Armband, luego se realiza el procesamiento de las señales electromiográficas EMG a través del Análisis Multirresolución de la Transformada Wavelet Discreta, con su wavelet madre Daubechies 4 y nivel de descomposición 2. Una vez procesadas las señales se extrae información útil, por medio de la extracción de características Valor Absoluto Medio (MAV), Raíz Cuadrática Media (RMS) y Longitud de la forma de Onda (WL) de cada uno de los subconjuntos de datos. Posteriormente, con las características obtenidas, se prepara el conjunto de datos para los modelos de aprendizaje supervisado seleccionados: red neuronal *Multi Layer Perceptron*, Árbol de decisión y Random Forest. El porcentaje de datos bien clasificados para los tres modelos es del 91%, 82% y 94%, respectivamente. El diseño de la interfaz de traducción es amigable con el usuario ya que consta de 4 botones claramente identificados con la función que cumplen.

**PALABRAS CLAVE:** <INGENIERÍA Y TECNOLOGÍA ELECTRÓNICA>, <INTELIGENCIA ARTIFICIAL>, <DISCAPACIDAD AUDITIVA>, <SENSORES MUSCULARES>, <SEÑALES ELECTROMIOGRÁFICAS (EMG)>, <TRANSFORMADA WAVELET DISCRETA>, <WAVELET MADRE DAUBECHIES 4>, <FPGA PYNQ-Z1 (SOFTWARE - HARDWARE)>.





## ABSTRACT

This research developed a system aimed to people with hearing disabilities who use the fingerprint alphabet for their communication. This device allows the translation of the conventional alphabet symbols represented in signs to letters using supervised machine Learning models. The implementation, processing and classification is executed on an embedded FPGA platform PYNQ-Z1, which is developed under the Linux open source system compatible with the Python program and its different bookstores. The process is initially developed with the adquisition of data through the myo armband muscle sensor. The electromiographic signals (EMG) are then processed through the Multiresolution Analysis of the Discrete Wavelet transformed with its mother wavelet Daubechies 4 and the level of decomposition 2. Once the signals are processed, useful information is extracted through the extraction of characteristics Mean Absolute Value (MVA), Root Mean Squared (RMS) and Waveform Length (WL) of each of the data subsets. Subsequently, with the characteristics obtained, the data set is prepared for the Learning models, supervised and selected: Multi Layer Perceptron neural network, Decision tree and Random Forest. The percentage of well-classified data for the three models is 91%, 82%, and 94% respectively. The design of the translation interface is user friendly as it consists of 4 buttons clearly identified with the function they fulfill.

## KEYWORDS:

ELECTRONIC ENGINEERING AND TECHNOLOGY / ARTIFICIAL INTELLIGENCE / HEARING IMPAIRMENT / MUSCLE SENSORS / ELECTROMIOGRAPHIC SIGNALS / TRANSFORMED DISCRETE WAVELET / WAVELET MOTHER DAUBECHIES 4 / FPGA PYNQ-Z1 (SOFTWARE-HARDWARE)



## INTRODUCCIÓN

En el Ecuador, el índice de personas con discapacidad auditiva según estadísticas del Consejo Nacional para la Igualdad de Discapacidades representan el 14,11% (62.921) de personas registradas en esta institución, el 44% son mujeres y 56% son hombres (CONADIS 2018). Alrededor del mundo hay aproximadamente 72 millones de personas sordas que usan 300 lenguas de señas.

Estas personas al no tener un síntoma visible de su discapacidad, presentan varios problemas para desenvolverse adecuadamente en la sociedad, solo se los advierte cuando se les habla, pero como casi nadie domina la lengua de señas no se puede establecer una comunicación inmediata.

Hay algunos métodos de intervención que se utilizan para la comunicación de personas con discapacidad auditiva dependiendo de su nivel de sordera, son: la Lectura labiofacial, el método verbotonar, la palabra complementada, la dactilología, la lengua de signos, etc; siendo esta última la más utilizada a nivel mundial; cabe mencionar que existen variaciones entre países y regiones de este método de comunicación (Martínez et al. 2008a).

Otro medio de comunicación es a través de un intérprete de lengua de señas, pero existe el riesgo de realizar interpretaciones erróneas con consecuencias graves; la lengua de señas, no solo son los movimientos de brazos y manos, también las expresiones faciales y corporales juegan un papel importante.

Entonces, el planteamiento para el desarrollo de la interfaz de traducción del alfabeto dactilológico, surge de la oportunidad de brindar una comunicación inmediata a las personas sordas, al ser ellos mismo quienes utilicen el dispositivo.

El desarrollo involucra una amplia investigación teórica, programación, implementación, además de la ejecución de pruebas para validar el funcionamiento. El hardware utilizado en la interfaz está conformado por sensores musculares, controlador y el software cumple con la función de procesamiento de señales, extracción de características, clasificación y visualización. La funcionalidad de la interfaz está limitada a la clasificación y traducción del alfabeto dactilológico, es decir, las letras del abecedario del alfabeto convencional en lengua de señas.

## ANTECEDENTES

Actualmente, se realizan varios trabajos para mitigar la brecha del lenguaje que tienen las personas con discapacidad auditiva y que hacen uso de la lengua de señas como medio de comunicación, mediante distintas formas, por ejemplo: detectar los movimientos de las manos con sensores (flexibles, acelerómetros, musculares bipolares, etc.), cámaras y visión artificial, electromiografía.

A nivel internacional la investigación realizada por Kosmidou y Hadjileontiadis en la que utilizan los datos de un sensor para electromiografía superficial de 5 canales y un acelerómetro de 3 dimensiones colocados en la mano dominante de la persona que usa lengua de señas, los mismos que fueron analizados usando Intrinsic Mode Entropy (IMEn) para el reconocimiento automático del lenguaje de signos griegos (GSL). Los resultados obtenidos al analizar 60 palabras del GSL con una repetición de 10 veces por 3 personas nativas del lenguaje de signos griego mostraron más del 93% de precisión en la clasificación promedio (Kosmidou y Hadjileontiadis 2009).

Además, en el trabajo (Pérez y Pável 2016) se plantea dos etapas, el reconocimiento de imágenes del lenguaje de señas mexicano, utiliza las cámaras del dispositivo Kinect con una frecuencia de muestreo de una imagen por segundo, cada una se compara con los patrones almacenados del sistema en la base de conocimientos, la etapa de aprendizaje inicia cuando detecta un nuevo patrón con una similitud inferior al 90% entre la imagen capturada y el patrón almacenado. El procedimiento se basa en una estimación de similitud.

El trabajo denominado “A Detail Study of Wavelet Families for EMG Pattern Recognition” los resultados sugieren que la transformada wavelet continua CWT, con una wavelet madre Sombrero mexicano en una escala 32 y Symlet 6 en una escala 16 son óptimas wavelets madres para la clasificación de señales EMG, mientras que con los coeficientes reconstruidos de la transformada wavelet discreta DWT con wavelet madre Daubechies 4 y Symlet 4 en un segundo nivel de descomposición son recomendados en el reconocimiento de patrones de señales EMG. También indican que la DWT ofrece un menor costo computacional y alta exactitud en la clasificación (Too et al. 2018).

En el trabajo desarrollado por Guzmán Arellano Diego Alejandro con el tema “Guante electrónico para traducir de lenguaje de señas a caracteres con voz artificial y conexión inalámbrica a dispositivos

móviles para personas con discapacidad Auditiva y de lenguaje en la Universidad Técnica de Ambato”, el sistema consta de un guante electrónico de 18cm x 8.5cm que aloja a los sensores flexibles, un brazalete ubicado en la muñeca en donde se encuentra un acelerómetro, tarjeta electrónica, pantalla TFT y el speaker. El guante tiene dos modos de traducción, el primero en que el usuario emite la traducción mediante el sintetizador de voz artificial y el segundo modo establece una conexión bluetooth con un dispositivo móvil, lo que permite recibir y transmitir mensajes visualizándolos en la pantalla del guante (Guzmán Arellano 2017).

En un trabajo titulado “Reconocimiento de caracteres del alfabeto dactilológico mediante redes neuronales artificiales: Un enfoque experimental” (Auquilla, Palacio-Baus y Saquicela 2015) se establece una metodología de cuatro etapas para el reconocimiento dactilológico, la primera es la adquisición de imágenes con una cámara de un teléfono celular conectada a Matlab mediante una dirección IP, posteriormente se realiza el tratamiento y procesamiento de la imagen convirtiéndola en escala de grises previo a la binarización, como penúltima etapa esta la clasificación de los patrones utilizando redes neuronales artificiales (RNA) multicapa entrenada con el algoritmo de retropropagación de error o *backpropagation* (Zurada, 1992), finalmente la presentación de resultados en pantalla de los caracteres reconocidos como una cadena. Este procedimiento obtuvo un porcentaje de acierto superior al 88%, las letras más complicadas de clasificar fueron la *C* y *W*, mientras que algunas de las letras menos propensas a equivocación fueron la *B* y la *I*.

Con el uso de las nuevas tecnologías y la visión artificial fue desarrollado el trabajo titulado, “Sistema para el reconocimiento del alfabeto dactilológico” el cual está dividido en varias etapas, la adquisición de la imagen, pre procesamiento que mejora la calidad de la imagen y segmentación, la extracción de características y finalmente la clasificación. Para la última etapa se utilizó *la familia de algoritmo de votación* (ALVOT), los resultados que obtuvieron es el 82% representa a los objetos bien clasificados, mientras que el 18% restante forma el conjunto de objetos mal clasificados.

## **JUSTIFICACIÓN**

### **Justificación Teórica**

En el Ecuador hay un gran porcentaje de personas que desconocen la lengua de señas convirtiéndose en un limitante en la comunicación y desarrollo de actividades cotidianas de las personas con discapacidad auditiva por lo cual se plantea el desarrollo de esta interfaz de traducción, como una solución a esta problemática, mejorar la calidad de vida y disminuir la brecha de comunicación.

También, el estado ecuatoriano en la Ley Orgánica de Discapacidades, artículo 70, manifiesta lo siguiente: “Se reconoce la lengua de señas ecuatoriana como lengua propia y medio de comunicación de las personas con discapacidad auditiva” (Registro Oficial 2012) y con base al Plan Nacional del Buen Vivir cuyo objetivo número dos plantea: “Auspiciar la igualdad, la cohesión, la inclusión y la equidad social y territorial, en la diversidad” (Plan Nacional del Buen Vivir, 2017), se promueven políticas que impulsan el acceso y uso de todas formas de comunicación: auditiva, visual, sensorial, y otras tecnologías que permitan la inclusión de las personas con cualquier discapacidad.

Analizando los dispositivos tecnológicos que se han desarrollado hasta el momento para la traducción o interpretación de la lengua de señas ecuatoriana, encontramos algunas limitaciones, por ejemplo, en los guantes de traducción, estos están diseñados para personas adultas y con ciertas características físicas, mientras que en el reconocimiento de señas utilizando cámaras y sistemas de visión artificial estas señas deben ser estáticas, la mano del usuario no debe moverse mientras se captura la imagen y tener ciertas condiciones de iluminación.

### **Justificación Aplicativa**

El proyecto es de carácter social, está dentro de las líneas de investigación de la Escuela Superior Politécnica de Chimborazo (ESPOCH), cumple con uno de los objetivos del Plan Nacional de Buen vivir ya mencionados; es de gran utilidad, puede ser instalado en entidades financieras, comerciales y educativas; facilitando el desenvolvimiento e inclusión de las personas con discapacidad auditiva.

La ejecución del proyecto será en varias etapas, inicialmente se seleccionará la posición adecuada de los sensores, una vez adquiridas las señales se realizará el procesamiento digital, para el reconocimiento de patrones se analizará la técnica de inteligencia artificial más efectiva, de esta forma de acuerdo a la seña realizada en la interfaz se mostrará el texto equivalente. Finalmente, se ejecutarán las pruebas de la interfaz de traducción, para verificar y corregir posibles errores.

Con las características que presenta el proyecto se puede mencionar que su realización es factible ya que se dispone de bibliografía especializada, recursos de software y hardware, el conocimiento adquirido para elaborar la interfaz y reconocimiento adecuado del alfabeto dactilológico.

## **OBJETIVOS**

### **Objetivo General**

Diseñar e implementar una interfaz para la traducción de caracteres del alfabeto dactilológico a palabras utilizando sensores musculares y reconocimiento de patrones.

### **Objetivos Específicos:**

- Investigar el estado del arte sobre dispositivos tecnológicos para la traducción de la lengua de señas reportados en literatura.
- Desarrollar la metodología adecuada para el procesamiento digital de las señales electromiografías discretizadas sobre una plataforma embebida.
- Diseñar el algoritmo de reconocimientos de patrones más óptimo para la clasificación y reconocimiento de las letras del alfabeto dactilológico.
- Implementar la interfaz de traducción de caracteres del alfabeto dactilológico utilizando una plataforma embebida.
- Verificar el funcionamiento de la interfaz de traducción a través de pruebas para determinar el porcentaje de acierto en el reconocimiento de patrones.

## **CAPÍTULO I**

### **1. MARCO TEÓRICO REFERENCIAL**

#### **1.1. Discapacidad Auditiva**

Se define como la dificultad que muestran algunas personas para desarrollar actividades cotidianas y de trabajo, percibir estímulos auditivos del medio ambiente y del lenguaje oral, como consecuencia de la pérdida de la capacidad para oír. Habilidades del habla, pensamiento, lenguaje, conducta, desarrollo social y emocional también son afectadas por causa de la deficiencia auditiva. Las personas pueden presentar pérdida auditiva desde el nacimiento o adquirirla al pasar los años (Consejo Nacional de Fomento Educativo 2010, p. 16-18 ; Ministerio de Educación Chile 2007, p. 7).

El acceso a información, desarrollo y comunicación entre personas con deficiencia auditiva y sujetos que hacen uso de la lengua oral como medio de comunicación, es de vital importancia, por lo que se han establecido algunos métodos y técnicas que ayuden a romper o disminuir la barrera del lenguaje. Dependiendo del grado de discapacidad auditiva o el que mejor se acople a la persona se hacen uso de estos métodos. (Martínez et al. 2008b, p. 220-222).

Dependiendo del grado de discapacidad existen algunos métodos de comunicación, entre los cuales tenemos los siguientes:

##### **1.1.1. *Métodos oralistas***

Estos métodos utilizan principalmente los movimientos de la boca y labios que se forman al decir un fonema, sílaba, palabra o frase. También el sentido de la vista y movimientos corporales, de manos y dedos facilita la comunicación de personas con deficiencia auditiva. Dentro de este método está la lectura labiofacial, método verbotonal, y la palabra complementada.

### **1.1.2. *Métodos gestualistas***

Este método abarca la lengua de signos y la dactilología, en los dos casos se realizan signos manuales y gestos corporales. La lengua de signos es la expresión y configuración gesto-espacial y percepción visual de la comunidad con deficiencia auditiva severa, no existe una lengua de signos universal, varía de acuerdo a la región, país e inclusive ciudad; en el Ecuador se encuentra establecida y reconocida a la Lengua de señas ecuatoriana (LSEC) como medio de comunicación de personas ecuatorianas con discapacidad auditiva (Oviedo, Carrera y Cabezas 2015).

La lengua de signos asocia un gesto de manos y movimiento corporal a palabras, frases, meses del año, números, alfabeto, etc. Es así que el presente trabajo utiliza el alfabeto dactilológico ecuatoriano como base de estudio.

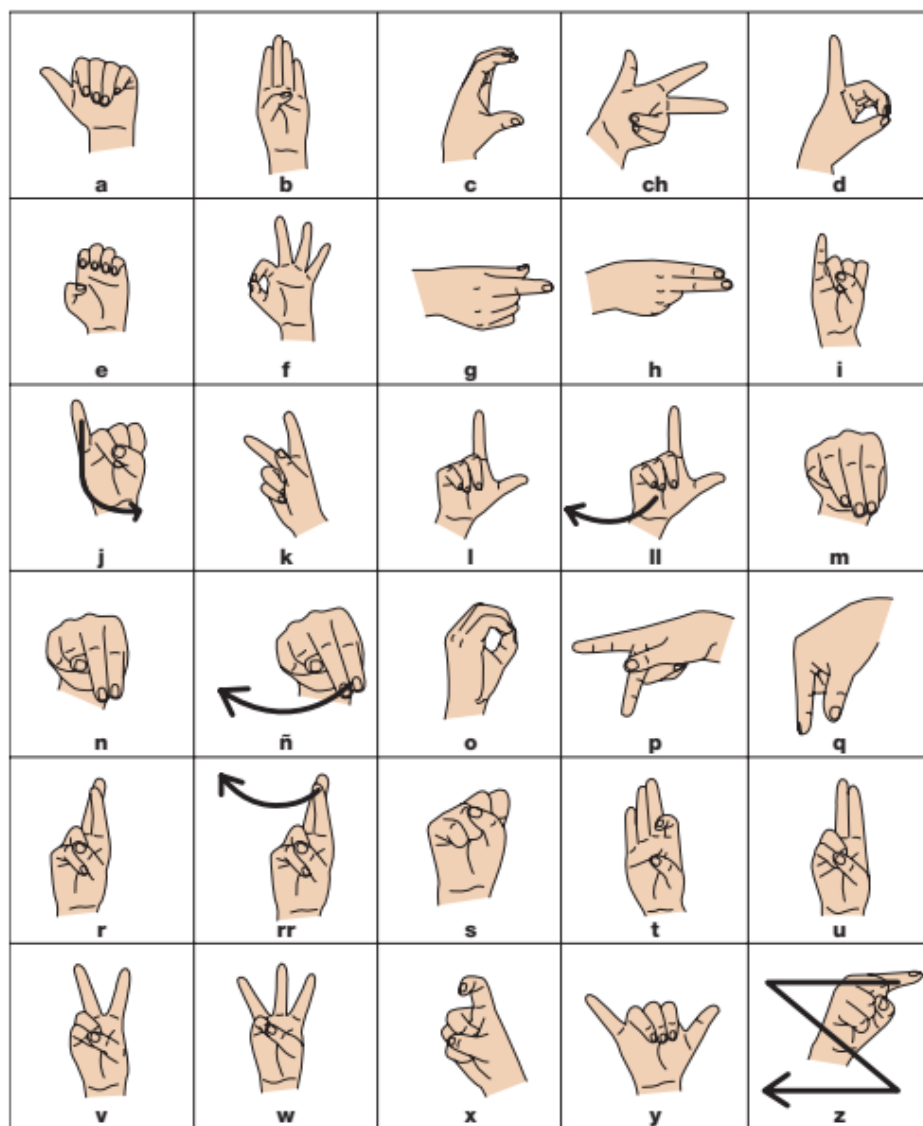
### **1.1.3. *Métodos Mixtos***

La comunicación bimodal, comunicación total y el bilingüismo forman parte de este método, son una combinación de los métodos oralistas y gestualistas, se pueden aplicar tanto a niños y adultos con pérdida auditiva y obtener buenos resultados de aprendizaje de la lengua oral, lectura y escritura.

## **1.2. *Alfabeto dactilológico***

Es una representación manual de cada letra del alfabeto de la lengua oral, así como existen varios idiomas también hay diferentes lenguas de señas con su respectivo alfabeto de acuerdo a cada país. La base de estudio del presente trabajo es el alfabeto dactilológico ecuatoriano mostrado en la figura 1-1 y que consta de 30 letras. Cada seña es realizada con una sola mano ya sea la izquierda o derecha dependiendo de la mano dominante de la persona que lo realice.





**Figura 1-1:** Alfabeto dactilológico ecuatoriano

**Fuente:** <http://www.sordosecuador.com/p/diccionarios-de-lengua-de-senas.html>

### 1.3. Electromiografía (EMG)

Estudia y registra la actividad eléctrica generada al contraer o relajar el músculo estriado, es decir, la diferencia de potencial producida por las fibras musculares que conforman la unidad motora al realizar los movimientos voluntaria o espontáneamente.

El valor registrado da el concepto de la actividad neuromuscular asociada con la contracción, pero no indica necesariamente la fuerza del músculo realizada, más bien, qué músculo del sistema nervioso

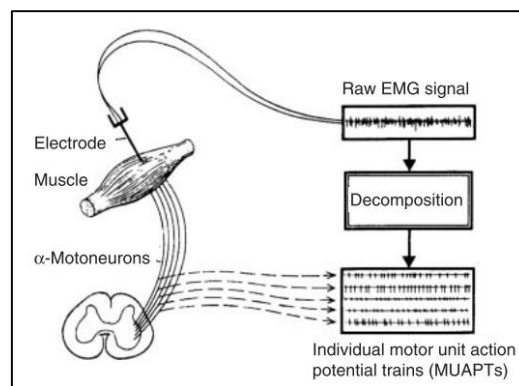
central se activa más, también indica indirectamente que músculo realiza una función protagónica al momento de ejecutar un movimiento específico (Caballero et al. 2002).

Existen dos tipos o técnicas de electromiografía: invasiva y superficial, la diferencia entre las dos está en la forma de adquisición de la señal y el rango de frecuencias de la misma. En general una señal EMG está en el rango de frecuencias entre los 0 – 10kHz y la amplitud de potencial eléctrico varía desde los micro voltios ( $\mu V$ ) a mili voltios(mV).

### 1.3.1. *Electromiografía invasiva*

Este método se denomina invasivo ya que penetra el músculo para registrar la actividad eléctrica generada por una unidad motora específica figura 2-1. La adquisición de la señal es a través del denominado electrodo de aguja, que consiste en una aguja delgada de metal, así, la amplitud de la señal obtenida varía de acuerdo al área de registro que ocupa el electrodo y la distancia que hay entre el electrodo y la unidad motora; mientras más cerca este el electrodo a la fuente de la señal la amplitud es mayor.

El límite de frecuencia de la señal EMG obtenida con un electrodo de aguja concéntrico (CNE) está por debajo de los 2kHz y con un electrodo para electromiografía de fibra única es de 10kHz (Merletti y Parker 2004, p. 43).



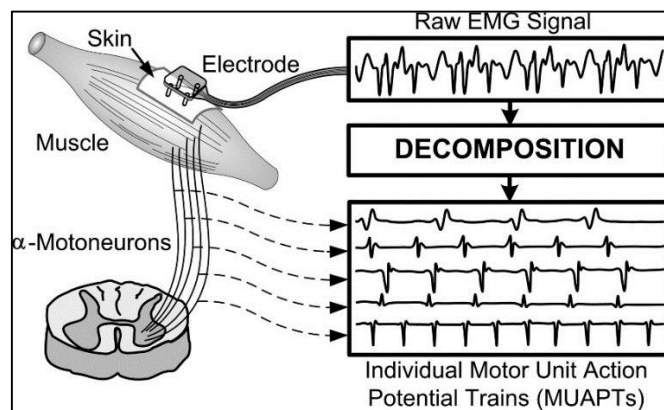
**Figura 2-1:** Electromiografía invasiva

**Fuente:** (Merletti y Parker 2004)

Este tipo de electromiografía es más utilizado en la medicina, para detectar lesiones o rehabilitación del área afectada, debido a la inserción del electrodo resulta dolorosa y requiere de supervisión médica.

### 1.3.2. Electromiografía de superficie (SEMG) o no invasiva

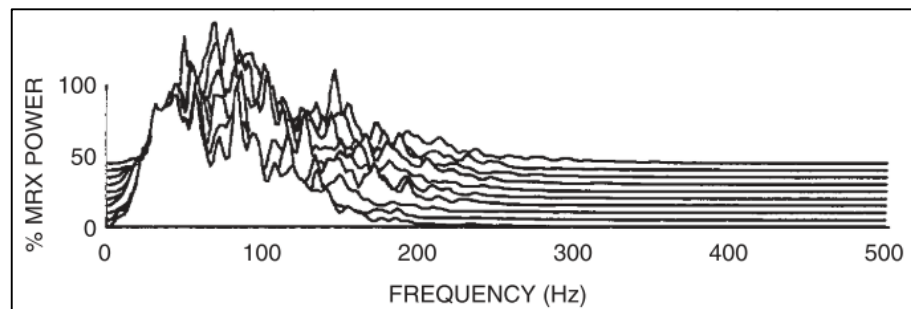
Este tipo de electromiografía utiliza electrodos superficiales para la adquisición y registro de la actividad eléctrica del musculo, estos son colocados sobre el tejido muscular sin producir dolor o lesión en el musculo de estudio. La señal registrada es de un grupo de unidades motoras que se encuentran bajo el área cubierta por el electrodo, como se puede apreciar en la figura 3-1. La electromiografía de superficie es ampliamente utilizada para el control de prótesis mioeléctricas, análisis de un gesto o para evaluar la fatiga muscular en los deportistas.



**Figura 3-1:** Electromiografía superficial, señal EMG

**Fuente:** (De Luca et al. 2006)

El contenido de frecuencia de las señales EMG superficiales está por debajo de los 500Hz, pero la mayor actividad electromiográfica oscila alrededor de los 50 y 150Hz como se puede apreciar en la figura 4-1 y la amplitud de potencial eléctrico está entre los 50 $\mu$ V y menos de los 10mV (Betancourt O., Giraldo Suárez y Franco B. 2004).



**Figura 4-1:** Espectro de frecuencias de varias señales EMG

**Fuente:** (Merletti y Parker 2004)

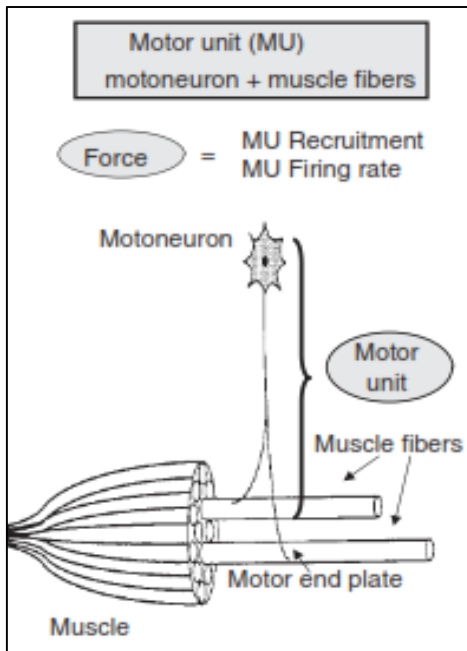
### **1.3.3. Bases fisiológicas**

El cuerpo humano es una máquina compleja, funciones como respirar, latir del corazón, contraer o relajar los músculos son a causa de impulsos eléctricos, estos impulsos son el resultado de la acción electroquímica de algunos tipos de células, las cuales generan un diferencial de potencial mediante partículas ionizadas (iones de potasio, calcio) y la despolarización de sus membranas celulares. Así, el sistema nervioso como centro de control del cuerpo humano, transporta y transmite la información sensitiva del organismo al cerebro para procesarla; esta información procesada es devuelta en forma de impulsos eléctricos a través de los nervios a las partes del cuerpo (Cameron y Skofronick 1978).

La neurona, unidad básica del sistema nervioso recibe y transmite impulsos eléctricos denominados también potenciales de acción, está compuesta por tres partes principales: el cuerpo principal denominado Soma, las dendritas que son extensiones celulares encargadas de recibir la información y otras extensiones celulares llamados axón encargados de enviar una respuesta. En el cuerpo humano están distribuidas tres tipos de neuronas: las neuronas sensoriales que están conectadas a receptores para así responder y detectar los estímulos externos, el segundo tipo son las neuronas motoras encargadas de controlar la actividad muscular (enviar y recibir impulsos eléctricos desde la espina dorsal hasta las fibras musculares) y el último tipo son las neuronas interneuronas encargadas de interconectar neuronas con la particularidad que no se conectan a receptores sensoriales ni fibras musculares. El envío y recepción de información entre neuronas es a través del proceso llamado sinapsis y la unión de las dendritas con el axón de las neuronas involucradas (Aldás Arias y Vanegas Jadán 2016).

El sistema músculo-esquelético está compuesta principalmente por neuronas motoras (motoneuronas), fibras musculares, músculos y el esqueleto, juntos proveen soporte al cuerpo y la capacidad de realizar movimientos. En el músculo podemos distinguir entre 2 unidades principalmente, la unidad anatómica y la unidad funcional; la primera es la llamada fibra muscular, mientras que la segunda recibe el nombre de unidad motora (Cifuentes González 2010).

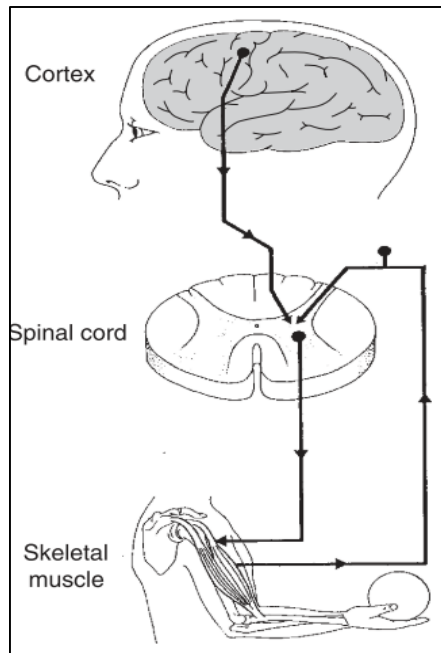
La unidad motora (UM) es la unidad fisiológica fundamental del sistema motor, está compuesta por una célula nerviosa (motoneurona) y por todas las fibras musculares que están inervadas por ella figura 5-1. El sistema nervioso como centro de control del cuerpo humano procesa el grado de contracción de las fibras musculares según el número de UM reclutadas y la frecuencia de activación de cada una (Osuna, Gonzalez y Ilzarbe 2013).



**Figura 5-1:** Componentes de la unidad motora

**Fuente:** (Merletti y Parker 2004)

Entonces, para realizar un movimiento se involucran neuronas sensoriales y motoras, el cerebro genera la orden en forma de impulso eléctrico que es transportado a través de la espina dorsal hasta llegar a las neuronas motoras y por medio de ellas a las fibras musculares del músculo que debe realizar la acción, ya sea de contracción o relajación figura 6-1.



**Figura 6-1:** Representación esquemática de los  
mecanismos básicos de control

**Fuente:** (Merletti y Parker 2004)

## 1.4. Sensores

Son dispositivos que detectan o captan señales físicas de algún fenómeno tales como temperatura, presión, luz, aceleración, distancia, actividad muscular, entre otras y las transforman en otro tipo de señales por lo general eléctricas.

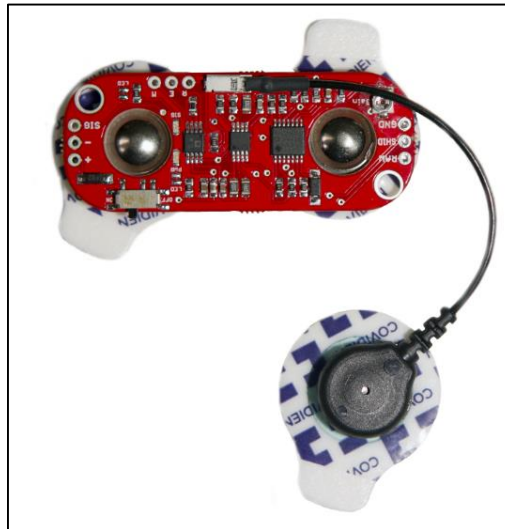
### 1.4.1. *Sensores musculares superficiales*

Este tipo de sensores captan y registran la actividad muscular producida por el grupo de unidades motoras que se encuentran en el área ocupada por los electrodos. Dependiendo del fabricante podemos encontrar sensores de uno o varios canales de recepción de la señal, con un electrodo extra que actúa de referencia de la señal, o sensores que entregan una señal con un procesamiento digital previo.

#### 1.4.1.1. *Sensor muscular MyoWare*

Este sensor es propiedad de Advancer Technologies utilizado para medir, filtrar y rectificar señales generadas de la actividad eléctrica muscular. La salida analógica varía de 0 a Vs voltios que puede ser adquirida por un microcontrolador con un convertidor analógico - digital (ADC). Cuando el grupo

muscular objetivo se contrae la tensión de salida del sensor aumenta. También cuenta con un potenciómetro para ajustar la relación exacta entre el voltaje de salida y la actividad muscular (Pololu Corporation 2014).

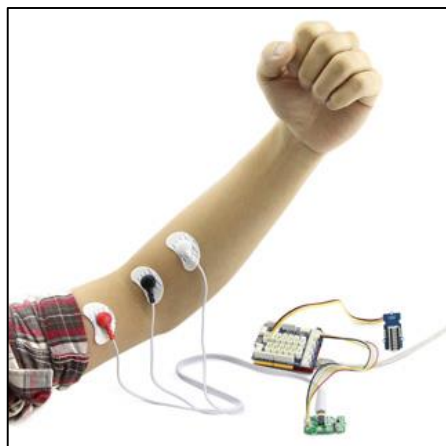


**Figura 7-1:** Sensor muscular MyoWare con electrodos

**Fuente:** <https://a.pololu-files.com/picture/0J6842.1200.jpg?079ca2904c5d27f9e41f71bbb3edd9a4>

#### *1.4.1.2. Sensor muscular EMG (Electromiografía)*

Este sensor de la marca Seeedstudio (Ver figura 8-1), sirve como puente para detectar los pequeños pulsos eléctricos producto de la actividad muscular y transformarlos en una señal reconocible por Arduino. En reposo el voltaje de salida es de 1.5V, cuando se detecta actividad muscular el voltaje puede subir hasta un máximo de 3.3V (BOTSciencie 2013).

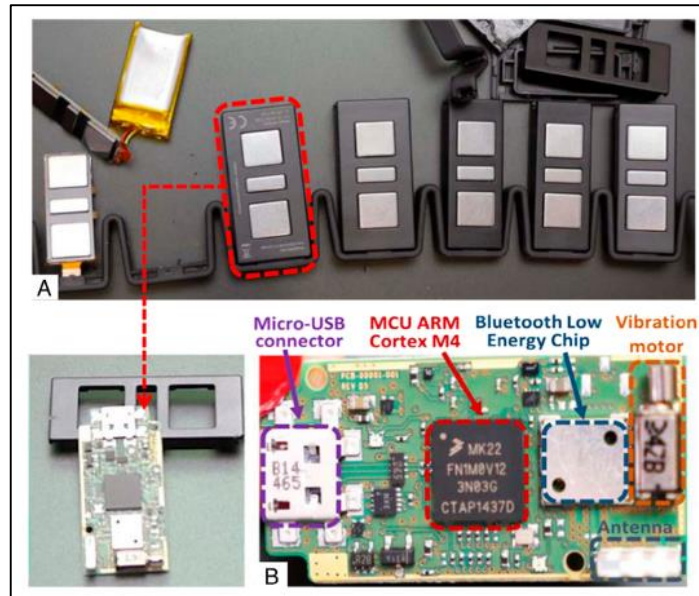


**Figura 8-1:** Sensor muscular EMG

**Fuente:** <http://www.seeedstudio.com/document/pics/EMG3.jpg>

#### 1.4.1.3. Myo armband

Es un brazalete de sensores desarrollado por la empresa canadiense Thalmic Labs, posee ocho sensores EMG de acero inoxidable de grado médico, una unidad de medición inercial (IMU) de nueve ejes, conexión Bluetooth para el envío de los datos inalámbricamente, procesador ARM Cortex M4, leds indicadores, retroalimentación con vibraciones cortas y largas, batería de litio recargable. En la figura 9-1 se puede apreciar la vista exterior de los electrodos y los elementos internos del módulo principal.



**Figura 9-1:** Myo Armband A. Vista externa B. Elementos que conforman el módulo principal

**Fuente:** (Visconti et al. 2018)

Cada módulo del brazalete contiene un sensor EMG, por tal motivo están enumerados del 1 al 8 para la identificación de las señales (Ver Figura 10-1).



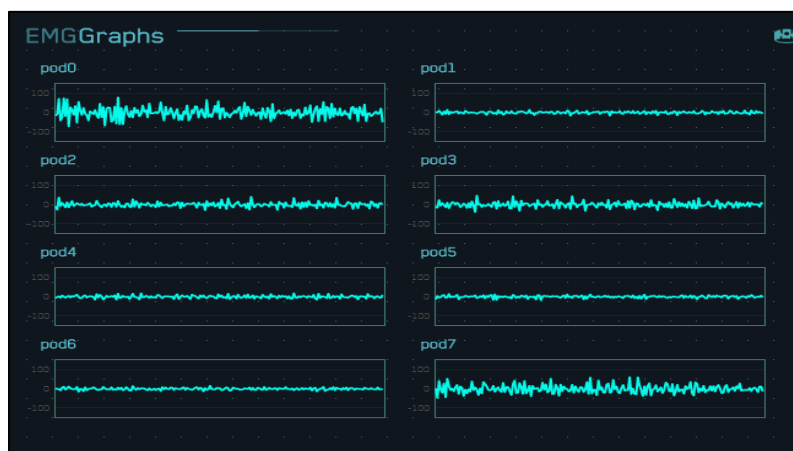
**Figura 10-1:** Ocho electrodos de superficie EMG en Myo Armband

**Fuente:** (Visconti et al. 2018)



A diferencia de otros sensores EMG cuyos valores de la señal mioeléctrica medida está en milivoltios, este dispositivo capta los potenciales eléctricos en unidades de activación, las mismas que son valores enteros comprendidas en el rango de -128 a 128. La frecuencia de muestreo de cada sensor EMG es de 200Hz y de la unidad de medición inercial es de 50Hz.

Por defecto el dispositivo reconoce cinco gestos: tic doble, puño, dedos abiertos, giro de palma de la mano a la izquierda o derecha; sin embargo, los datos sin procesar o raw data también pueden ser obtenidos de cada sensor (Ver Figura 11-1) y también hay que tener en cuenta que el brazalete Myo ha sido diseñado para funcionar mejor en la parte más ancha del antebrazo, es decir, antebrazo superior (Abduo y Galster 2015).



**Figura 11-1:** Raw data de cada sensor EMG

**Fuente:** <https://developerblog.myo.com/myocraft-logging-imu-and-raw-emg-data/>

#### 1.4.2. *Análisis comparativos de sensores musculares*

La tabla 1-1, muestra las características más relevantes de los sensores disponibles en el mercado para el desarrollo del presente trabajo de investigación, con estos detalles determinaremos el ideal para la adquisición de las señales EMG al momento de realizar una seña del alfabeto dactilológico ecuatoriano.

**Tabla 1-1:** Comparación de características entre sensores musculares superficiales

Sensor		MyoWare	Sensor muscular EMG	Brazalete Myo Armband
Fuente de energía	Externa	X		
	Integrada		X	X

Tipo de señal	EMG pura (raw)	X	X	X
	EMG procesada	X		
Procesador	Externo	X	X	
	Integrado			X
Comunicación	Inalámbrica			X
	Serial	X	X	
Número de electrodos – sensores		2	2	8
Electrodo de referencia a tierra		X	X	

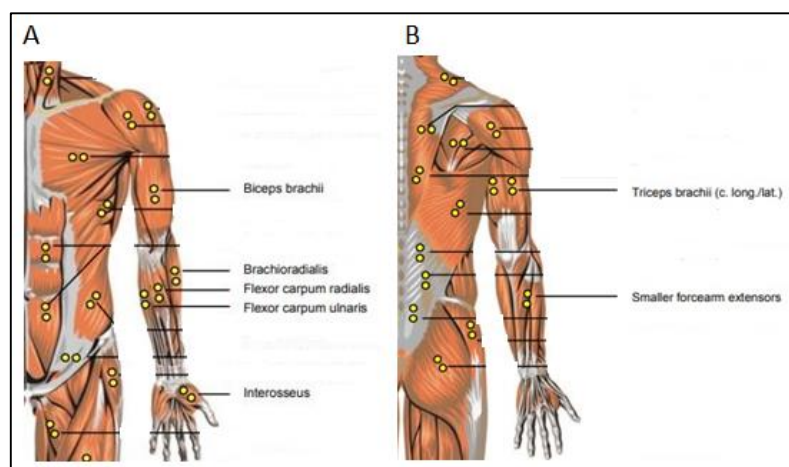
Realizado por: Ñacato Ana, 2019

### 1.5. Sistema de posicionamiento de electrodos

Para el registro de señales electromiográficas existen varias recomendaciones de trabajos relacionados de sistemas de posicionamiento de electrodos, en los que mencionan y resaltan el proyecto europeo SENIAM (Surface Electromyography for the Non-invasive Assessment of Muscles).

SENIAM, es un proyecto cuyos trabajos están relacionados a la electromiografía de superficie (SEMG), su objetivo es dar recomendaciones y los procedimientos para la ubicación y colocación de sensores, procesamiento de señales y simulación de modelos (Hermens et al. 1999).

En la figura 12-1 se muestra la ubicación de los electrodos superficiales para las áreas de estudio de brazos y manos.



**Figura 12-1:** Ubicación de electrodos superficiales en brazos y manos.

A. Vista frontal, B. Vista Posterior

Fuente: (Konrad 2006)

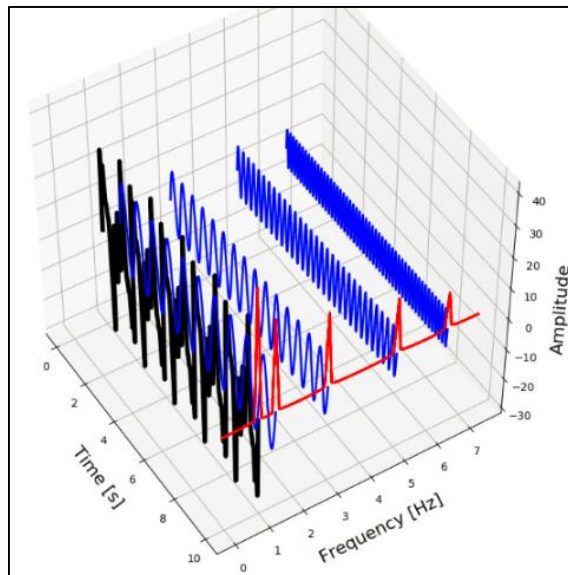
## 1.6. Procesamiento de señales EMG

Una señal cuyas propiedades no varían en el tiempo se conocen como señales estacionarias, dentro de estas señales están las señales periódicas. Las señales que no se repiten a lo largo del tiempo son conocidas como señales no-estacionarias.

El procesamiento de señales analiza, representa, transforma, manipula y obtiene información que no es fácilmente disponible o extraíble de las señales originales, todo esto a través de transformaciones matemáticas.

### 1.6.1. Transformada de Fourier

La transformada de Fourier TF, es una herramienta matemática que se basa en el principio de que cualquier señal puede ser representada por una ecuación conformada por funciones seno y coseno. En la figura 13-1 tenemos un ejemplo de como la transformada de Fourier descompone una señal desde una representación del dominio de tiempo (tiempo-amplitud) al dominio de la frecuencia (frecuencia-amplitud). En color negro está la señal compuesta por varias señales de diferentes frecuencias, en azul las componentes sinusoidales y en rojo la señal en el dominio de la frecuencia, donde vemos la frecuencia y amplitud de las señales componentes.

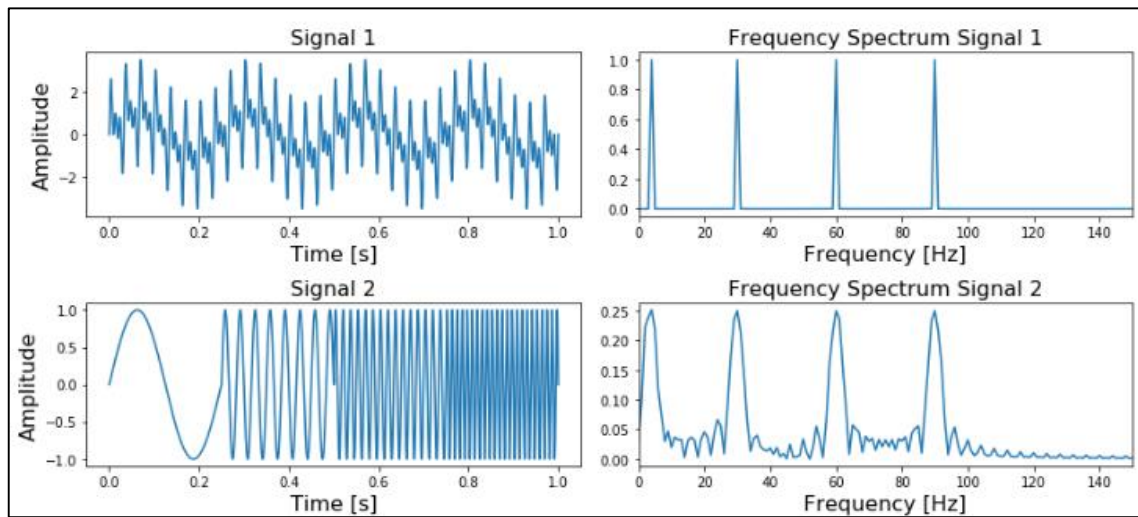


**Figura 13-1:** Transformación del dominio del tiempo al dominio de la frecuencia de una señal

**Fuente:** [http://ataspinar.com/wp-content/uploads/2018/01/signals3D\\_4.png](http://ataspinar.com/wp-content/uploads/2018/01/signals3D_4.png)

Estos datos de frecuencia-amplitud pueden resultar bastante útiles dependiendo de la aplicación. Pero hay una característica de que debemos tener en cuenta, al realizar la transformación del dominio de tiempo al dominio de la frecuencia, perdemos la información referente al tiempo, es decir, cuando o en qué momento está presente una determinada frecuencia.

A continuación, en la figura 14-1 presentamos dos señales con cuatro componentes de frecuencia (4, 30, 60 y 90Hz), en la parte superior vemos una señal periódica que se repite a lo largo del tiempo junto con su espectro de frecuencias; en la parte inferior está otra señal, pero con la diferencia de que, la primera parte de esta señal se encuentra a 4Hz, la segunda parte a 30Hz, tercera a 60Hz y la última parte a 90Hz. El espectro de frecuencias está al lado derecho.



**Figura 14-1:** Señales con su espectro de frecuencias para Señal estacionaria vs señal no estacionaria

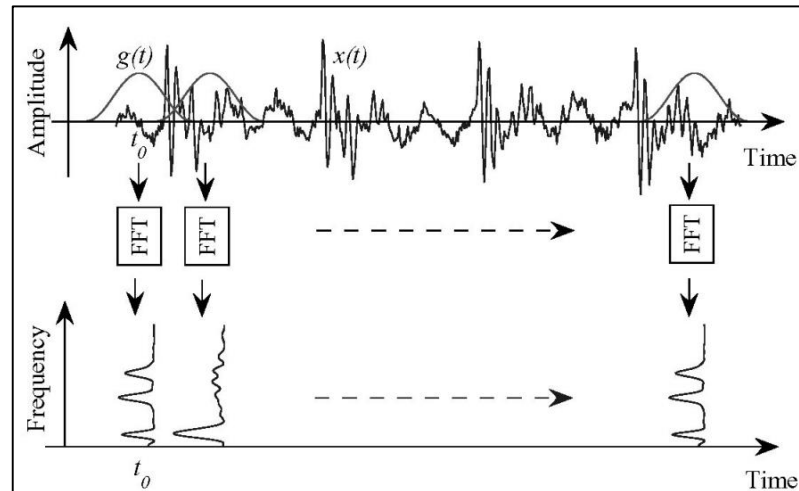
**Fuente:** [http://ataspinar.com/wp-content/uploads/2018/08/fft\\_spectrum-1.png](http://ataspinar.com/wp-content/uploads/2018/08/fft_spectrum-1.png)

En el espectro de frecuencia de las dos señales vemos que tienen los 4 picos, pero la información que se ha perdido es en donde o cuando están presentes cada frecuencia. Con este ejemplo, vemos que la Transformada de Fourier es óptima para un análisis en señales estacionarias aun cuando hay una pérdida de información, pero no sucede lo mismo para señales no-estacionarias. Sin embargo, existen un gran número de señales con las características de no periodicidad, entre ellas están señales Electromiográficas, electrocardiogramas, encefalogramas, entre otras.

### 1.6.2. Transformada de Fourier de Tiempo Corto

La transformada de Fourier de tiempo corto fue planteada por científicos como una herramienta para resolver la desventaja que presenta la Transforma de Fourier de resolución en tiempo. Esta

transformada consiste en dividir la señal original  $x(t)$  en partes iguales o también conocidas como función ventana deslizante  $g(t)$ , a cada porción de señal se aplica la Transformada de Fourier hasta cubrir la totalidad de la señal, como resultado tenemos las componentes de frecuencias de la señal de cada parte analizada, ver figura 15.1.

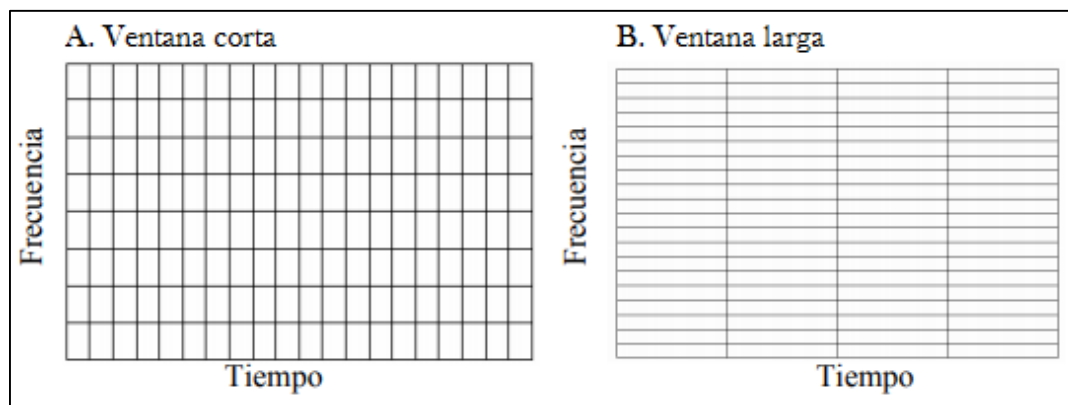


**Figura 15-1:** SFTF en una señal  $x(t)$

**Fuente:** (Gao y Yan 2006)

Al aplicar la SFTF tenemos información de tiempo y frecuencia de una señal, pero una vez determinado el tamaño de ventana  $g(t)$  este es fijo para todas las frecuencias, así entramos al límite teórico propio de la Transformada de Fourier que es el Principio de incertidumbre.

- Cuanto más pequeño sea el tamaño de la ventana, más sabremos dónde se ha producido una frecuencia en la señal, pero menos sobre el valor de la frecuencia en sí. Figura 16-1.A
- Cuanto más grande sea el tamaño de la ventana, más sabremos sobre el valor de la frecuencia y menos sobre el tiempo. Figura 16-1.B



**Figura 16-1:** Tamaño de la ventana y su influencia en la resolución tiempo-frecuencia

**Fuente:** (Zamorano Castaño 2010)

Por lo mismo, gran parte de las señales necesitan un análisis más flexible, es decir que podamos variar el tamaño de la ventana para conseguir mayor precisión tiempo – frecuencia.

### 1.6.3. *Transformada Wavelet*

La Transformada Wavelet es una herramienta matemática poderosa, aplicada en el análisis de señales no estacionarias, sísmicas, de sonido, en el procesamiento de imágenes, reconocimiento de patrones, detección de anomalías sintomáticas en medicina, entre otras. Al igual que la SFTF su estudio se basa en la división de la señal original en ventanas, pero el tamaño de estas ventanas es variable, una ventana angosta para altas frecuencias y una ventana ancha para bajas frecuencias.

La TW traslada una señal del dominio espacial al dominio en escala mediante el uso de un grupo infinito de funciones base con energía finita, llamada Wavelets.

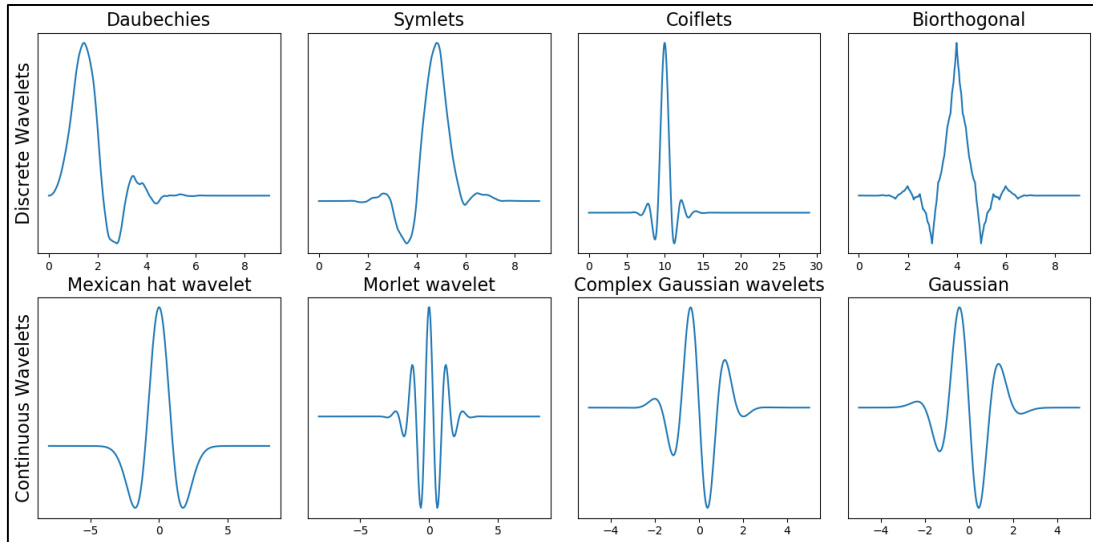
Las Wavelets son grupos de familias de funciones definidas en el espacio, empleadas como funciones de análisis, examinando la señal de interés en el plano tiempo-frecuencia para obtener sus características periódicas y no periódicas (Zamorano Castaño 2010). Una familia de Wavelet está definida por:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi \left( \frac{t-b}{a} \right) \quad (1)$$

$$a, b \in \mathbb{R}; a > 0$$

Donde,  $a$  es la variable de escala o tamaño de la wavelet,  $b$  es la traslación en el tiempo y  $\psi(t)$  la Wavelet Madre que genera cada uno de los componentes de la familia wavelet.

En la figura 17-1, observamos algunas de las principales familias Wavelets. Cada una tiene una forma, suavidad y dependiendo de la aplicación pueden ser útiles algunas y otras no.



**Figura 17-1:** Se observa algunas de las principales familias wavelets

Realizado por: Ñacato Ana, 2019

#### 1.6.3.1. Transformada Wavelet Continua CWT

Matemáticamente la Transformada Wavelet Continua está definida como:

$$\text{CWT}(a, b; \psi) = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} f(t) \cdot \psi^* \left( \frac{t-b}{a} \right) dt \quad (2)$$

En donde:  $f(t)$  señal a la que aplicamos la CWT

$\psi_{a,b}^*(t)$  Conjugado complejo de la función wavelet  $\psi_{a,b}(t)$

La señal transformada CWT, es una función de dos variables,  $a$  y  $b$ , factor de escala y de traslación respectivamente. Dichos factores son continuos, es decir que puede haber una cantidad infinita de wavelets. Podemos escalar la wavelet madre con un factor de  $a = 1.3$ , o  $1.31, 1.311, 1.3111$  etc.

A continuación, presentamos los conceptos de escala y traslación.

- **Escala**

En el análisis wavelet, este concepto es similar a la escala en un mapa geográfico. Cuando usamos una escala grande, podemos ver la señal como un todo y cuando usamos una escala pequeña, podemos obtener una vista detallada. Se asocia con la frecuencia de la misma manera, la baja frecuencia (alta escala) corresponde a la información global de la señal que normalmente cubre toda la señal, y la alta

frecuencia (escalas bajas) corresponde a información detallada de una característica oculta de la señal que ocurre en un tiempo relativamente corto. Los valores altos de escala se relacionan con señales dilatadas y un valor de pequeña escala corresponde a una señal comprimida. La relación escala frecuencia está definida como:

$$F_a = \frac{F_c}{a \cdot \Delta} \quad (3)$$

Donde:  $a$  es la escala

$\Delta$  es la frecuencia de muestreo (Hz)

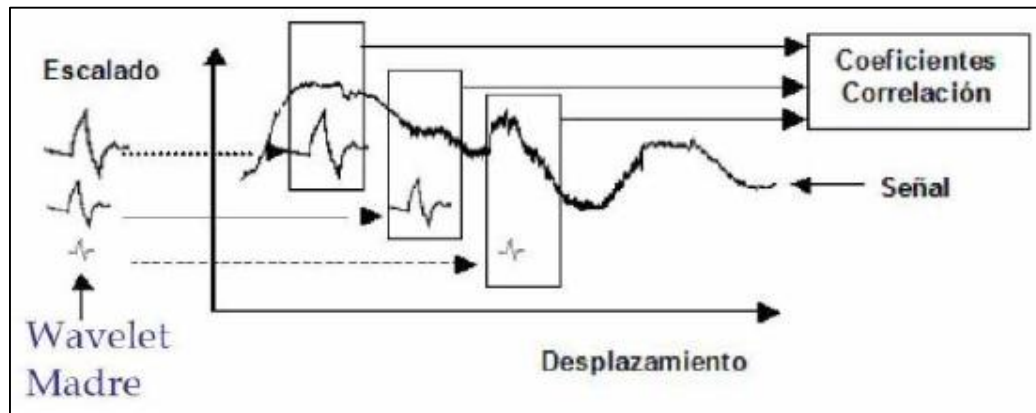
$F_c$  Frecuencia central de la función wavelet (Hz)

$F_a$  pseudo-frecuencia correspondiente a la escala  $a$  (Hz)

- **Traslación**

El factor de traslación corresponde al desplazamiento de la función wavelet a lo largo de la señal en el dominio del tiempo.

La traslación se realiza sobre el dominio del tiempo, mientras que el escalado se realiza conforme a la dilatación y compresión de ventanas, obteniendo coeficientes de correlación. La figura 18-1 muestra una representación esquemática de la aplicación de la CWT.



**Figura 18-1:** Aplicación de la CWT a una señal

**Fuente:** (Zamorano Castaño 2010)

### 1.6.3.2. Transformada Wavelet Discreta DWT

La DWT se basa en la discretización de los parámetros de escala y de tiempo de la CWT. Se utiliza para analizar y descomponer señales. El costo computacional es menor en comparación con la CWT. Matemáticamente está definida como:



$$\text{DWT}(j, k) = \frac{1}{\sqrt{a}} \int f(n) \psi(a^{-j}n - kb) dn \quad (4)$$

$$a = 2^j; b = k2^j \quad a > 0; b > 1 \quad (j, k) \in \mathbb{Z}^2$$

Dónde:  $a$  y  $b$  se definen generalmente según la escala diádica

$j$  representa el nivel de descomposición

$k$  es el factor de traslación

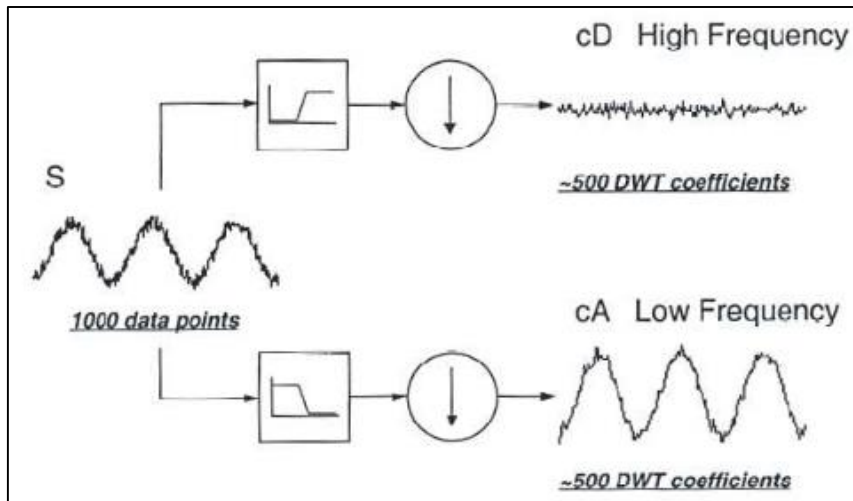
$f(n)$  la señal a analizar

- **Filtración: aproximaciones y detalles**

La señal producida por la voz humana tiene componentes de baja y alta frecuencia, si eliminamos los componentes de alta frecuencia la voz suena diferente, pero aún somos capaces de entender lo que se dice; por otro lado, si eliminamos un número de componentes de baja frecuencia el habla ya no sería entendida. Es por eso que, en algunas señales, las componentes de baja frecuencia representan la parte más importante, lo que aporta la identidad de la señal; mientras que el contenido de alta frecuencia puede ser considerado como secundario.

Por lo antes dicho, en el análisis Wavelet a menudo habla de aproximaciones (A) y detalles (D). Los componentes de gran escala y baja frecuencia de la señal están en las aproximaciones y los componentes de pequeña escala y alta frecuencia son los detalles de la señal.

En la figura 19-1 observamos el proceso de descomposición de la señal original (S), pasa a través de dos filtros complementarios y obtenemos dos señales. La señal con coeficientes de detalle son en su mayoría ruido de alta frecuencia, mientras que los coeficientes de la señal de aproximación contienen mucho menos ruido del que contenía en la señal original.



**Figura 19-1:** Diagrama esquemático de la descomposición de la señal original ( $S$ ) en una señal de alta frecuencia (coeficientes de detalle) y una señal de baja frecuencia (Coeficientes de aproximación)

Fuente: (Zamorano Castaño 2010)

- **Análisis Multirresolución**

Este Análisis se implementa como un banco de filtros, es decir, una cascada de filtros pasa alto y pasa bajo. Los bancos de filtros son una forma eficiente de dividir una señal en varias sub-bandas de frecuencia.

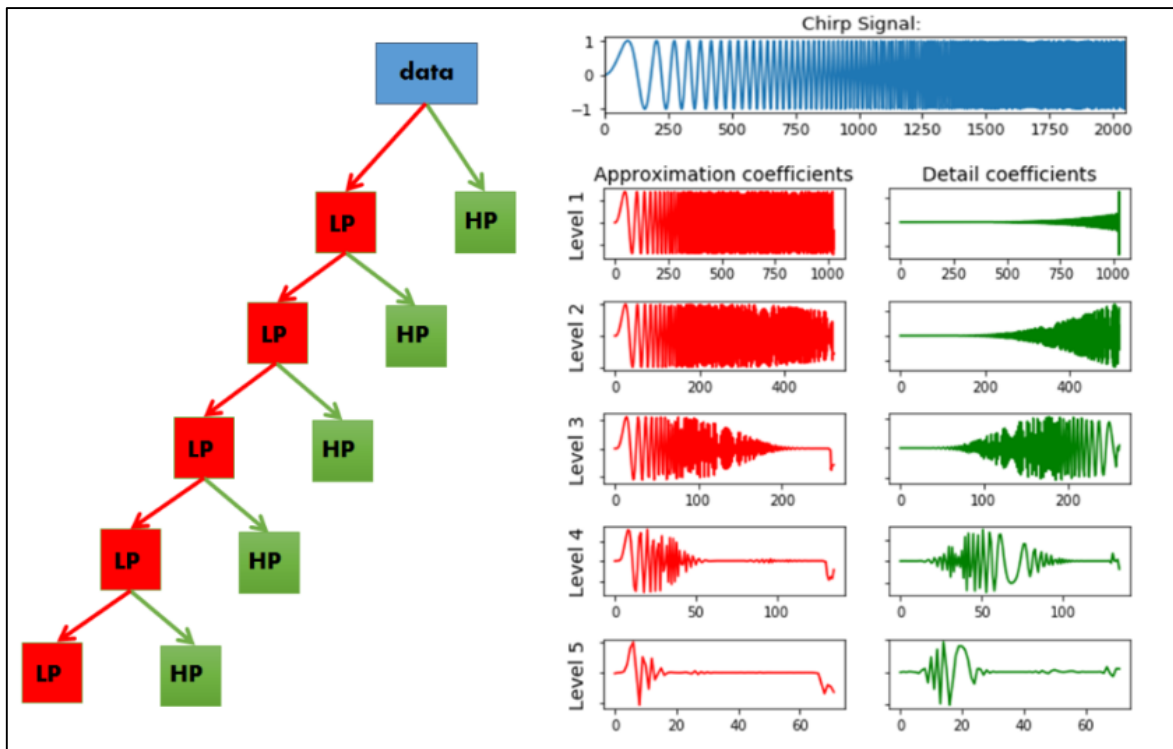
Para aplicar el DWT en una señal, comenzamos con la escala más pequeña. Como hemos visto antes, las escalas pequeñas se corresponden con frecuencias altas. Esto significa que primero analizamos el comportamiento de alta frecuencia. En la segunda etapa, la escala aumenta con un factor de dos (la frecuencia disminuye con un factor de dos), y estamos analizando el comportamiento alrededor de la mitad de la frecuencia máxima. En la tercera etapa, el factor de escala es cuatro y estamos analizando el comportamiento de la frecuencia alrededor de una cuarta parte de la frecuencia máxima, así hasta alcanzar el nivel máximo de descomposición.

**Nivel máximo de descomposición:**

En cada etapa posterior, el número de muestras en la señal se reduce con un factor de dos. Con valores de frecuencia más bajos, necesitará menos muestras para satisfacer la velocidad de Nyquist, por lo que no es necesario mantener el mayor número de muestras en la señal; solo hará que la

transformación sea computacionalmente costosa. Debido a esta disminución de la resolución, en algún momento del proceso, el número de muestras en nuestra señal será menor que la longitud del filtro wavelet y habremos alcanzado el nivel máximo de descomposición (Ahmet Taspinar 2018).

Por ejemplo, analicemos una señal con una frecuencia de hasta 1000 Hz. En la primera fase, dividimos la señal en una parte de baja frecuencia y una parte de alta frecuencia, es decir, 0-500 Hz y 500-1000 Hz. La segunda fase, dividimos la parte de baja frecuencia en dos partes: 0-250 Hz y 250-500 Hz. Para la tercera etapa, dividimos la parte de 0-250 Hz en una parte de 0-125 Hz y de 125-250 Hz. Así, continuamos hasta alcanzar el nivel de refinamiento deseado o cuando se agote la muestra. Ver figura 20-1.



**Figura 20-1:** Descomposición wavelet de nivel 5, a una señal con frecuencia de 1000Hz, de color rojo tenemos los coeficientes de aproximación y de color verde los coeficientes de detalle.

**Fuente:** (Ahmet Taspinar 2018)

Para un nivel de descomposición  $n$ , obtenemos un subconjunto de coeficientes de detalle  $cd_1, cd_2, \dots, cd_n$  y subconjunto de coeficientes de aproximación  $ca_1, ca_2, \dots, ca_n$

### Reconstrucción de la señal

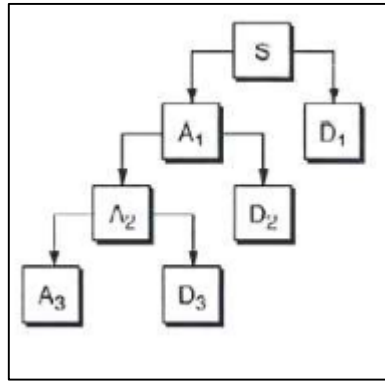
La reconstrucción de la señal resulta de aplicar la transformada inversa wavelet (IDWT) y obtenemos los subconjuntos de datos de aproximación  $A_1, A_2, \dots, A_n$  y subconjuntos de detalle  $D_1, D_2, \dots, D_n$ . Figura 21-1.

Para un nivel 1 la señal original resulta de:  $S = A_1 + D_1$

Nivel 2:  $S = A_2 + D_2 + D_1$

Nivel 3:  $S = A_3 + D_3 + D_2 + D_1$

Nivel n:  $S = A_n + D_n + \dots + D_2 + D_1$



**Figura 21-1:** Coeficientes de reconstrucción

**Fuente:** (Zamorano Castaño 2010)

#### 1.6.4. *Pywavelet*

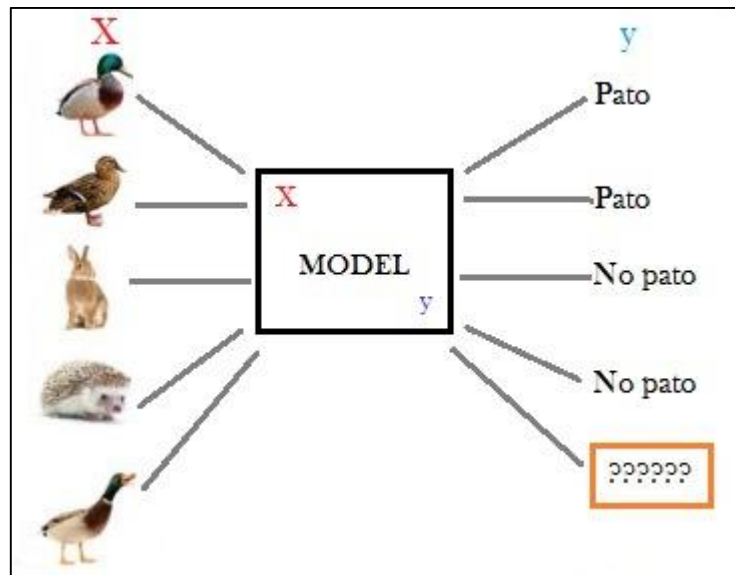
PyWavelets es un software para aplicar la Transformada Wavelet, de código abierto para Python. Combina una interfaz simple de alto nivel, con lenguaje de bajo nivel C y el rendimiento de Cython (Lee et al. 2019).

Nos ofrece las opciones de obtener:

- DWT e IDWT para una, dos dimensiones y n dimensiones.
- Un análisis multinivel de la DWT e IDWT
- CWT
- SWT
- Paquete de descomposición y reconstrucción wavelet para 1 dimensión y dos dimensiones. Entre otras opciones.

## 1.7. Aprendizaje Supervisado

Este tipo de aprendizaje se basa en descubrir la relación existente entre unas variables de entrada y unas variables de salida, es decir, el aprendizaje surge de enseñarles a estos algoritmos el resultado que deseamos obtener para un determinado valor o conjunto de características, después de mostrarle muchos ejemplos si se dan las condiciones el algoritmo será capaz de predecir un resultado correcto incluso cuando le mostramos valores de entrada que no ha visto antes. Lo mencionado podemos observarlo en la figura 22-1.



**Figura 22-1:** Aprendizaje supervisado

Realizado por: Ñacato Ana, 2019.

Dentro de este tipo de aprendizaje, hay dos tipos de algoritmos: de Regresión y Clasificación. Si el objetivo es predecir salidas de valor continuo, nos presentamos con un caso de regresión, mientras que si el objetivo es predecir salidas de valor discreto tenemos un caso de clasificación. Por ejemplo, si predecimos el valor de una casa, tenemos un problema de regresión, por otro lado, para predecir si un correo electrónico es spam o no, entonces nos encontramos frente a un problema de clasificación.

### 1.7.1. Modelos de Clasificación

La elección entre dos clases lo denominamos clasificación binaria, para una elección entre más de dos clases es una clasificación multiclase.

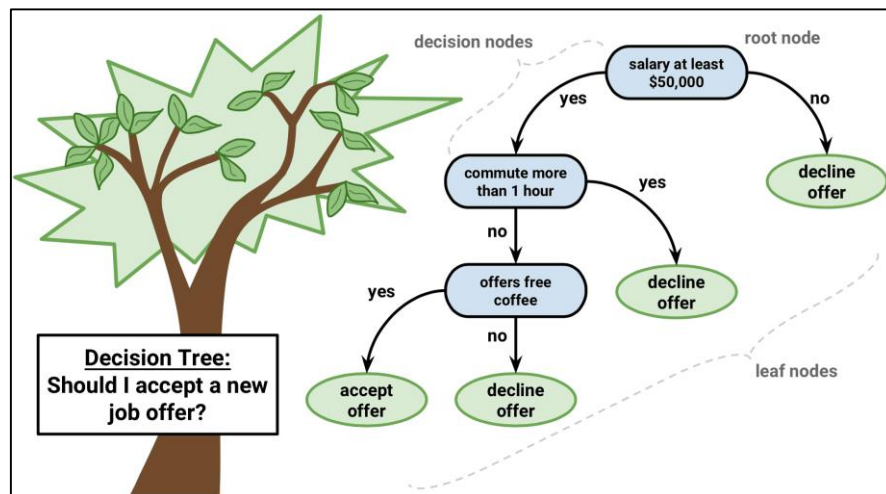
Algunos de los tipos de algoritmos de clasificación son:

- Árbol de decisión clasificación – Decision Tree Classifier
- Bosques aleatorios clasificación – Random Forest Classifier
- Redes Neuronales – Neural Network
- Regresión logística – Logistic Regression
- Vecinos más cercanos – K Nearest Neighbors Classifier (KNN)
- Máquinas de vectores de soporte – Support Vector Machine (SVC)

#### 1.7.1.1. Árbol de decisión

Este modelo se basa en una estructura similar a la forma de un árbol (figura 23-1), está compuesto por un nodo raíz, nodos de decisión y nodos de resultado (hojas).

En el nodo raíz empieza el análisis de clasificación de una clase, luego se descomponen en los atributos de entrada que toman el nombre de nodos de decisión. El nodo de decisión, es uno de los atributos que tienen dos o más ramas que representan los posibles valores que puede tomar el atributo, es decir, aquí se realizan elecciones basadas en las características del caso a clasificar. El nodo de respuesta o nodos terminales, muestra la clase a la que pertenece el caso, resultado de las series de decisiones



**Figura 23-1:** Árboles de decisión

Fuente: <https://bookdown.org/content/2031/images/terminology.png>

Uno de los parámetros que se debe tomar en cuenta para obtener buenos resultados es la profundidad máxima del árbol, ya que, si es muy alta el clasificador tiene a memorizar los resultados, lo que buscamos es que los clasificadores generalicen.

### 1.7.1.2. Random Forest

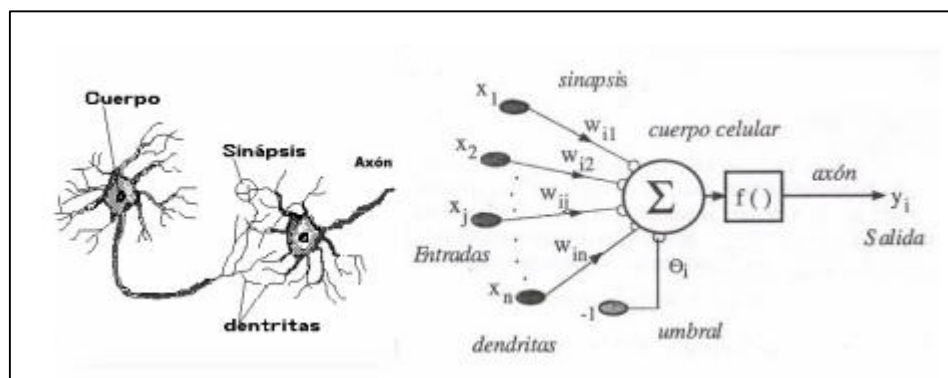
Este clasificador combina diversos arboles de decisión, la salida de cada árbol cuenta como un voto y la salida más votada es la respuesta de este clasificador.

Su nombre de aleatorio, se debe a que hay la posibilidad de seleccionar las características y el número de estas, para cada árbol creado. Esto le brinda flexibilidad para poder obtener gran variedad de árboles. El proceso del algoritmo es el siguiente:

- Creamos arboles de decisión seleccionando  $k$  características de las  $m$  totales del conjunto de datos ( $k < m$ ).
- Creación de diferentes arboles de decisión variando la cantidad de  $k$  y también se puede variar la cantidad de datos para estos árboles.
- Cada árbol predice la misma clase y le asignamos un voto, la clase que obtenga la mayoría de votos es considerada la salida final del clasificador de los bosques.

### 1.7.1.3. Redes Neuronales Artificiales

Son modelos matemáticos inspirados en el comportamiento del cerebro humano. Se crea una red con diferentes capas interconectadas para procesar la información. Cada capa está formada por un grupo de nodos o neuronas que transmite la información a los otros nodos de las capas siguientes. El modelo de una sola neurona artificial se puede entender en términos muy similares al modelo biológico. Tal como se muestra en la figura 24-1. Un diagrama de red dirigido define una relación entre las señales de entrada recibidas por las dendritas (variable  $x$ ) y la señal de salida (variable  $y$ ). Al igual que con la neurona biológica, la señal de cada dendrita se pondera (valores  $w$ ) de acuerdo a su importancia. Las señales de entrada son sumadas por el cuerpo de la célula y la señal se transmite de acuerdo con una función de activación indicada por  $f$  (Martínez 2018)



**Figura 24-1:** Red neuronal biológica vs red neuronal artificial

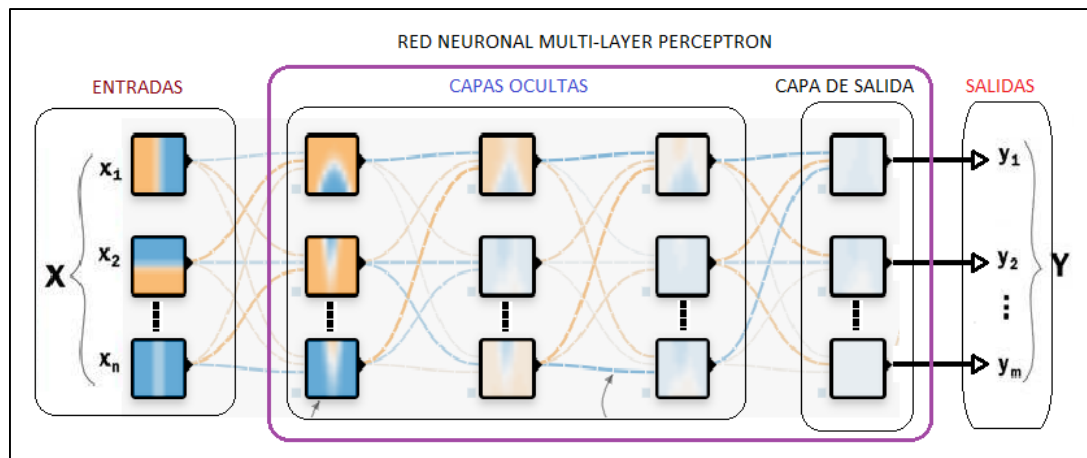
**Fuente:** (Martínez 2018)

De acuerdo a la aplicación, topología, algoritmo de aprendizaje pueden existir un sin número de redes neuronales, una de ellas es el Multi Layer Perceptron.

### Red Multi Layer Perceptron

El perceptron multicapa (MLP) surge como una extensión del perceptron simple para mejorar su rendimiento. Se caracteriza porque sus neuronas se agrupan en capas de diferentes niveles como se aprecia en la figura 25-1. Principalmente se distinguen tres capas: Capas de entrada, capas ocultas y capas de salida.

La capa de entrada aloja neuronas que cumplen la función de recibir al conjunto de características, señales o patrones del exterior, posteriormente propagan la información recibida a las siguientes capas. En las capas ocultas están las neuronas que realizan el procesamiento no lineal de los datos recibidos. Finalmente, en la última capa están las neuronas de salida que proporcionan el resultado de la red para cada uno de los datos de entrada.



**Figura 25-1:** Red neuronal Multi capa

Realizado por: Ñacato Ana, 2019

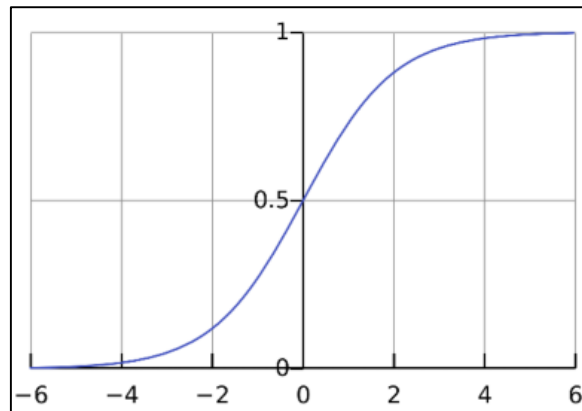
- Elementos principales:
  - a) Estructura de la red: Incluye el número de neuronas en la capa de entrada, el número de capas ocultas junto con sus neuronas y la capa de salida.
  - b) Función de activación: Una neurona biológica puede estar activa (excitada) o inactiva (no excitada); es decir, que tiene un “estado de activación”. Las neuronas artificiales también tienen diferentes estados de activación; algunas de ellas solamente dos, al igual que las biológicas, pero otras pueden tomar cualquier valor dentro de un conjunto determinado. La función activación calcula el estado de actividad de una neurona; transformando la entrada



global en un valor (estado) de activación, cuyo rango normalmente va de (0 a 1) o de (−1 a 1). Esto es así, porque una neurona puede estar totalmente inactiva (0 o −1) o activa representada por 1 (Matich Damián 2001).

La función sigmoid, es una de las funciones más utilizadas, está definida por la ecuación:

$$h_{\theta}(x) = \frac{1}{1 + e^{-\theta^T x}} \quad (5)$$



**Figura 26-1:** Función sigmoid

**Fuente:** (Aldás Arias y Vanegas Jadán 2016)

- c) Algoritmos de aprendizaje: El aprendizaje es un proceso fundamental de las RNA y de forma general, consiste en el ajuste de todos los parámetros de la red, en base a la actividad para la que se quiera utilizar dicha red. Partiendo de un conjunto de pesos aleatorios, el aprendizaje busca un conjunto de pesos que permita a la RNA, desarrollar una tarea determinada. El aprendizaje es un proceso iterativo, en el que la red, va refinando sus parámetros, para alcanzar el objetivo de la aplicación concreta que se intenta abordar. Las redes MLP utilizan una función de error que mide su rendimiento actual, en función de sus pesos. El aprendizaje se convierte en un proceso de búsqueda de aquellos pesos que hagan mínima dicha función (Pérez Formigó 2010).

### 1.7.2. Evaluación de modelos

La fase de clasificación aún no termina mientras no evaluemos los modelos seleccionados y entrenados. Con esto medimos la fiabilidad y precisión del modelo para clasificar, utilizamos los datos de prueba para comprobar si el modelo es capaz de predecir correctamente la clase a la que

pertenece cada conjunto de datos. A continuación, algunas de las principales métricas de clasificación: Matriz de confusión, Precisión, Recall también conocida como sensibilidad o tasa positiva real (TPR), Especificidad o tasa negativa real (TNR), F1-Score.

#### 1.7.2.1. Matriz de confusión

Es una tabla que puede ser usada para medir el rendimiento de un algoritmo de Machine Learning, principalmente en el Aprendizaje supervisado, ya sea en una clasificación binaria o multiclase.

Cada fila de la matriz representa las salidas verdaderas (actual) y las columnas representan las salidas predichas (predicted) o viceversa. Nota: Es importante conocer como están dispuestas nuestras filas y columnas para evitar errores o confusiones.

En la tabla 2-1 representa la matriz de confusión de una Clasificación Binaria:

**Tabla 2-1:** Matriz de confusión para una clasificación binaria

		Predicted	
True		Negativo	Positivo
	Negativo	TP Verdadero Positivo	FP Falso Positivo
	Positivo	FN Falso negativo	TN Verdadero Positivo

Realizado por: Ñacato Ana, 2019.

En donde:

**Verdadero Positivo (TP):** Cuando la clase verdadera del punto de datos era 1 (positivo) y la predicha también es 1 (positiva)

**Verdadero Negativo (TN):** Cuando la clase verdadera del punto de datos era 0 (negativa) y la predicha también es 0 (negativa)

**Falso Positivo (FP):** Cuando la clase verdadera del punto de datos era 0 (negativo) y la predicha es 1 (positiva)

**Falso Negativo (FN):** Cuando la clase verdadera del punto de datos era 1 (positivo) y la predicha es 0 (negativa)

**Accuracy:** Es el porcentaje total de elementos clasificados correctamente. Esta medida no siempre es la más adecuada para datos desbalanceados. Por ejemplo, tenemos 1000 muestras, de las cuales 995 son casos negativos y 5 positivos. Supongamos que nuestro clasificador está diseñado para clasificar cualquier dato que se presente como negativo. El accuracy será del 99,5% a pesar de que el clasificador no podría reconocer ninguna muestra como positiva.

$$AC = \frac{TN + TP}{TN + FP + FN + TP} \quad (6)$$

**Precisión:** Proporciona información sobre el rendimiento con respecto a los falsos positivos.

$$Precision = \frac{TP}{FP + TP} \quad (7)$$

**Recall:** Brinda información del rendimiento del clasificador con respecto a los falsos negativos (cuántos fallaron).

$$Recall = \frac{TP}{FN + TP} \quad (8)$$

#### **Clasificación Multiclase:**

La matriz de confusión es una matriz cuadrada de  $n \times n$  donde  $n$  es número de salidas o etiquetas del clasificador. Los elementos de la diagonal principal representan el número de puntos para los cuales la salida predicha es igual a la salida verdadera, mientras que los elementos fuera de la diagonal son aquellos que están mal etiquetados por el clasificador.

Si los valores de los elementos de la diagonal son altos, indican que hay muchas predicciones correctas.

**Precisión:** Es la división entre los casos donde el algoritmo predijo correctamente la clase  $i$ , para la suma de todos los casos predichos como la clase  $i$  (correcta o incorrectamente)

$$Precision = \frac{M_{ii}}{\sum_{j=1}^n M_{ij}} \quad (9)$$

**Recall:** es la división entre los casos que el algoritmo predijo correctamente la clase  $i$ , para todos los casos que están etiquetados como  $i$ .

$$\text{Recall} = \frac{M_{ii}}{\sum_{j=1}^n M_{ji}} \quad (10)$$

**F1- Score:** Podemos interpretarlo como un promedio ponderado de la precisión y recall, donde un puntaje F1 alcanza su mejor valor en 1 y el peor puntaje en cero.

En general, tanto para la clasificación binaria o multiclase debemos tomar en cuenta lo siguiente:

- Altos valores para precisión y recall muestran que el clasificador está devolviendo resultados precisos (alta precisión), así también regresa la mayor parte de todos los resultados positivos (alta recall).
- Un sistema con alta recall, pero baja precisión devuelve muchos resultados, pero la mayoría de sus etiquetas predichas son incorrectas en comparación con las etiquetas de entrenamiento.
- Un sistema con alta precisión, pero baja recall es todo lo contrario, devuelve muy pocos resultados, pero la mayoría de sus etiquetas predichas son correctas en comparación con las etiquetas de entrenamiento.
- Un sistema ideal con alta precisión y alta recall devolverá muchos resultados, pero con todos los resultados etiquetados correctamente.

## 1.8. Sistemas Embebidos

Es un sistema electrónico, diseñado para realizar un determinado número de funciones, con frecuencia en tiempo real. Al contrario de lo que sucede con las computadoras que cumplen un propósito general, ya que están diseñadas para satisfacer una amplia gama de necesidades, mientras que los sistemas embebidos pueden satisfacer necesidades específicas. Los sistemas embebidos generalmente en una de sus partes tienen una computadora con características especiales, llamada microcontrolador, que se convierte en el cerebro del sistema. Este es solo un microprocesador que contiene una interfaz de entrada / salida en el mismo chip. Por lo general, estos sistemas tienen una interfaz externa para monitorear el estado y realizar diagnósticos del sistema.

Un sistema embebido consta de un microprocesador y un software que se ejecuta en él. Sin embargo, este software requiere un lugar para guardar y ejecutar el procesador. Esto puede ser denominada como memoria RAM o ROM, de la cual cierta cantidad es usada por el sistema embebido. Las

características principales de los sistemas embebidos es que son de bajo costo y bajo consumo de energía (Infotec 2016).

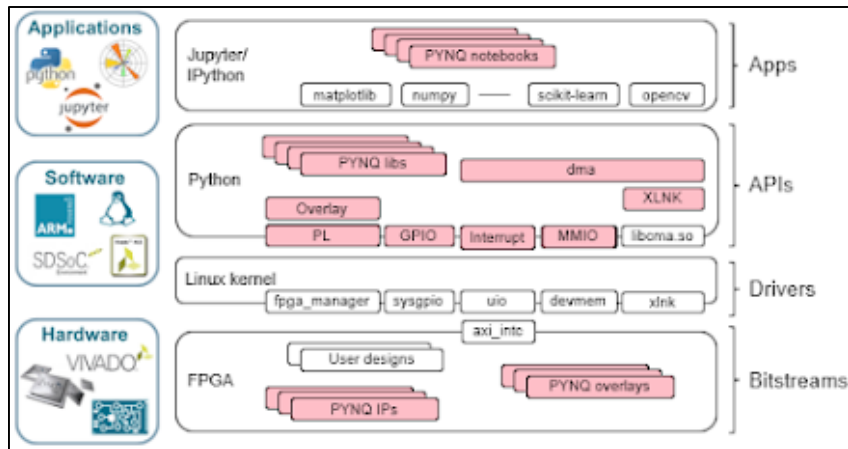
### **1.8.1. *Plataforma embebida – FPGA***

FPGA (Field Programmable Gate Arrays) es una serie de dispositivos basados en semiconductores a base de matrices de bloques lógicos configurables o CLB, donde además se conectan a través de lo que en el sector se denomina como interconexiones programables. Su principal característica y ventaja es que pueden ser reprogramados para un trabajo específico o cambiar sus requisitos después de haberse fabricado.

Normalmente la programación de los FPGA se realiza en lenguajes de programación de bajo nivel llamados Verilog o VHDL. Ambos sirven para “describir” al FPGA la forma en cómo debe manejar el hardware del mismo. Sin embargo, muchas empresas ofrecen sistemas de programación de más alto nivel que facilitan esta tarea aún más.

#### **1.8.1.1. *Digilent PYNQ-Z1***

Esta placa desarrollada por la empresa Digilent, está diseñada para usarse con PYNQ (figura 27-1), un nuevo framework de código abierto que permite a los programadores de sistemas embebidos explotar las capacidades de los SoC (Sistema en chip) programables Xilinx Zynq sin tener que diseñar circuitos lógicos programables. Este SoC se programa usando Python y el código se desarrolla y prueba directamente en el PYNQ-Z1. Los circuitos lógicos programables se importan como bibliotecas de hardware y se programan a través de sus API esencialmente de la misma manera que las bibliotecas de software se importan y programan.



**Figura 27-1:** Framework de código abierto PYNQ

Fuente: <http://lh3.googleusercontent.com/gCWTcM18HaJ-b->

En la tabla 3-1, se presenta algunas de las características de la plataforma PYNQ-Z1.

**Tabla 3-1:** Características FPGA PYNQ-Z1

	<b>PYNQ-Z1</b>
<b>Sistema en chip</b>	Zynq Z7020 (ARM A9 CPU)
<b>Memoria</b>	512MB DDR3
<b>Almacenamiento</b>	MicroSD
<b>Video</b>	HDMI In & Out
<b>Audio</b>	PDM integrated Mic, 3.5 mm PWM audio jack
<b>Puerto de red</b>	10/100/1000 Ethernet
<b>Expansion</b>	USB host (PS)
<b>GPIO</b>	1x Arduino Header
	2x Pmod
	16x GPIO pins
<b>Otras I/O</b>	6x LEDs
	4x Botones
	2x Dip switch
<b>Dimensiones</b>	87mm x 122mm

Realizado por: Ñacato Ana, 2019.

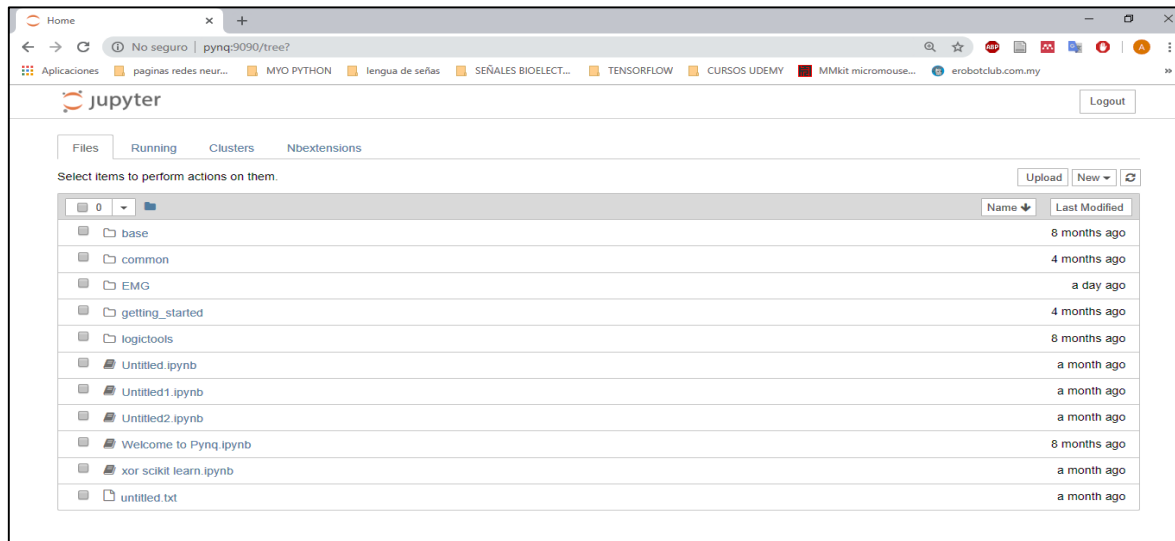
El software que se ejecuta en las CPU ARM A9 incluye:

- Servidor web que aloja el entorno de diseño de Jupyter Notebooks
- El núcleo y paquetes de IPython
- Linux
- Librerías base de hardware y API para FPGA

También se puede usar las herramientas Xilinx Vivado WebPACK ampliar el sistema base mediante la contribución de nuevas bibliotecas de hardware.

## Jupyter Notebook

Es un entorno informático interactivo basado en navegador. Se pueden crear documentos de Jupyter Notebook que incluyen código en tiempo real, widgets interactivos, diagramas, texto explicativo, ecuaciones, imágenes y video. En la figura 28-1 se muestra la página de Jupyter Notebook para PYNQ.



**Figura 28-1:** Jupyter Notebook para PYNQ-Z1

**Realizado por:** Ñacato Ana, 2019.

Una placa Zynq habilitada para PYNQ se puede programar fácilmente en Jupyter Notebook usando Python. Con Python, los desarrolladores pueden usar bibliotecas de hardware y superposiciones en la lógica programable. Las bibliotecas o superposiciones de hardware pueden acelerar el software que se ejecuta en una placa Zynq y personalizar la plataforma y las interfaces de hardware. El framework PYNQ que será instalada en la Micro-SD arranca sobre Linux e incluye el paquete PYNQ de Python y otros paquetes de código abierto.

## CAPÍTULO II

### 2. MARCO METODOLÓGICO

En este capítulo se presentan los métodos aplicados y las etapas de trabajo efectuadas para el desarrollo de la interfaz de traducción de caracteres del alfabeto dactilológico a palabras utilizando sensores musculares y reconocimiento de patrones (ver figura 1-2).

Para el desarrollo del proyecto, se inicia con el método sintético e inductivo en la etapa investigativa. El método sintético permite estructurar las ideas con el análisis y síntesis de la información recopilada de temas relacionados, mientras que el método inductivo facilita la generalización partiendo de premisas particulares.

El método heurístico es aplicado en la etapa de selección e implementación, ya que se estudia y analiza el desafío planteado, de tal forma que se obtenga una posible solución para la interfaz de traducción. Finalmente, el método experimental es utilizado para el análisis e interpretación de las señales de los sensores musculares y en la publicación de los resultados.



**Figura 1-2:** Diagramas de bloques del sistema

**Realizado por:** Ñacato Ana, 2019.

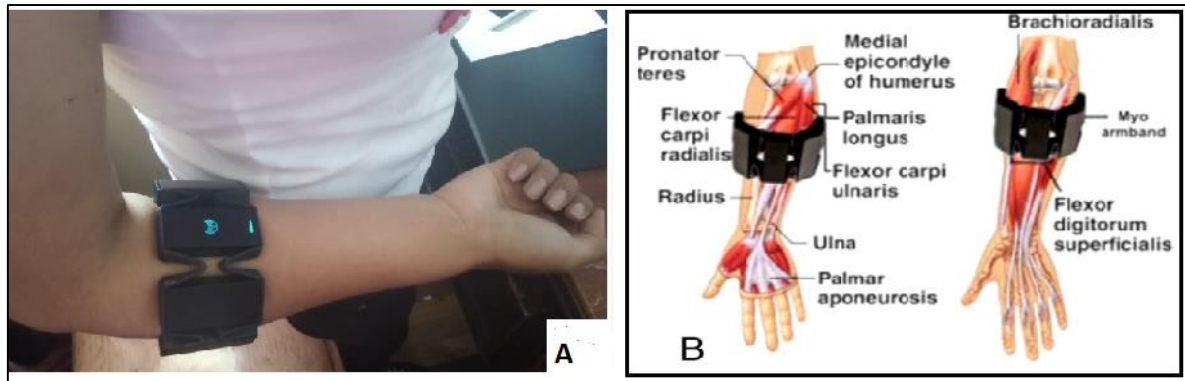
#### 2.1. Adquisición de señales EMG

El sensor muscular es de vital importancia para este proyecto, por tal motivo se determina que el idóneo para este trabajo es el brazalete Myo armband, ya que posee ocho electrodos de captación de señales EMG, incluye una batería de una duración aproximada de 3 a 4 horas, fácil uso y ubicación.



### 2.1.1. Ubicación de sensor

La representación manual de las señas del alfabeto dactilológico utiliza brazos y manos, nos centraremos y estudiaremos las recomendaciones dadas para estas regiones; es así que los sensores con los 8 electrodos se ubican alrededor del antebrazo, en los músculos Flexor radial del carpo, flexor superficial de los dedos; debemos procurar colocar el módulo principal con el ícono del brazaletes en este último músculo. Ver figura 2-2.

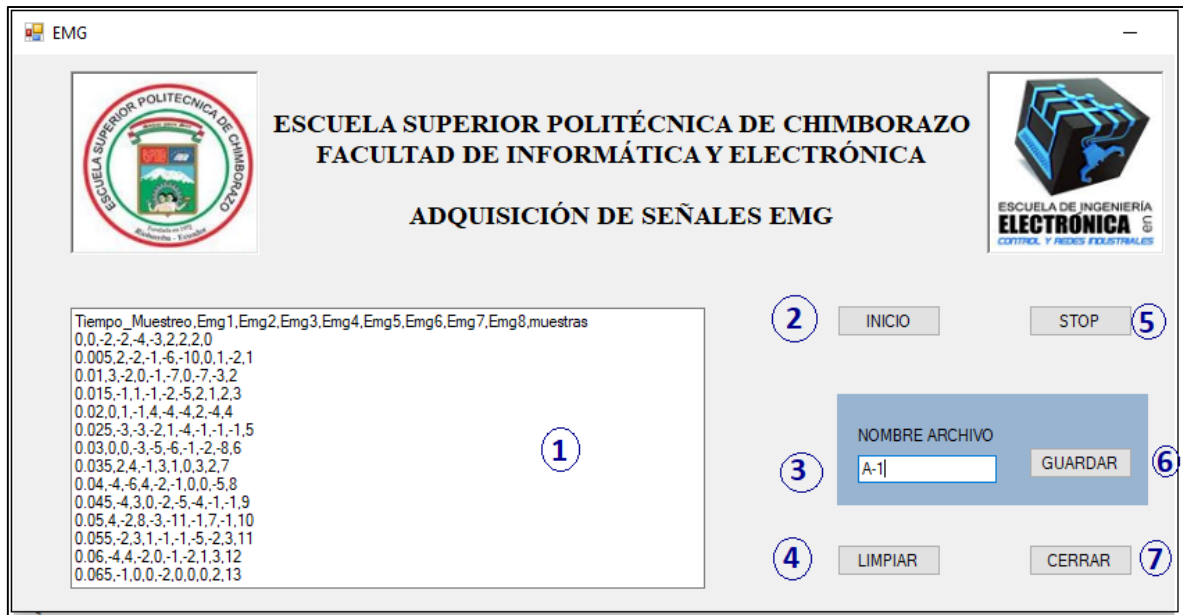


**Figura 2-2:** Sensor Myo, A. Antebrazo derecho, B. Músculos del antebrazo y Myo Armband

**Fuente:** Ñacato Ana, 2019

### 2.1.2. Software de adquisición

Para la adquisición de las señales a través de una interfaz desarrollada en Visual Studio, se puede crear y guardar un archivo CSV con cada uno de los datos adquiridos por los electrodos al realizar cada seña del alfabeto. Figura 3-2. El tiempo de adquisición de los datos es por 5 segundos, es decir, cada archivo generado contiene 1001 filas y 10 columnas. La primera fila contiene las etiquetas de cada columna, el resto corresponde a los valores de activación adquiridos en cada electrodo.



**Figura 3-2:** Interfaz de adquisición de señales EMG

**Realizado por:** Ñacato Ana, 2019

En donde:

1. Espacio en que se muestran los datos del sensor, en el siguiente orden: tiempo de muestreo, valores del electrodo 1, electrodo 2, electrodo 3, electrodo 4, electrodo 5, electrodo 6, electrodo 7, electrodo 8 y el número de datos adquiridos.
2. Con el botón INICIO se empieza la adquisición de datos
3. Espacio donde se escribe el nombre del archivo de acuerdo a la seña realizada.
4. Este botón limpia el espacio 1 y 3, para una nueva adquisición.
5. El botón de STOP, para la adquisición de los datos.
6. Al presionar este botón se guarda y crea un nuevo archivo CSV en la carpeta “bin\Debug” de la carpeta raíz.
7. Botón para cerrar la interfaz de adquisición.

En la figura 4-2 se observa una captura realizada al archivo CSV generado, con sus respectivas etiquetas y distribución en las celdas.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Tiempo_Muestreo	Emg1	Emg2	Emg3	Emg4	Emg5	Emg6	Emg7	Emg8	muestras					
2	0	2	-5	0	4	15	11	8	10	0					
3	0.005	-17	-14	-10	-16	-17	-10	-8	-5	1					
4	0.01	3	-2	-18	-1	4	-3	-1	-1	2					
5	0.015	20	11	-9	6	8	14	9	16	3					
6	0.02	1	-10	-4	-20	-9	-1	0	2	4					
7	0.025	-8	0	14	0	-11	-12	-19	-15	5					
8	0.03	7	12	6	6	6	7	3	3	6					
9	0.035	17	6	12	-4	-6	0	-4	-2	7					
10	0.04	-9	-2	5	2	-3	-16	-14	-13	8					
11	0.045	13	3	-5	3	-5	-6	-2	-3	9					
12	0.05	18	16	-16	46	55	44	39	30	10					
13	0.055	-10	-13	-10	-16	-12	-7	-6	-6	11					
14	0.06	-10	2	-7	5	8	7	5	-1	12					
15	0.065	-12	-3	-9	-16	-8	1	0	3	13					
16	0.07	1	1	3	0	9	-2	-7	0	14					
17	0.075	-7	-18	23	-39	-54	-43	-42	-30	15					
18	0.08	15	14	7	12	3	-1	-1	8	16					
19	0.085	-2	-7	4	-15	-12	-1	-3	-6	17					
20	0.09	-7	-6	8	6	-3	-14	-18	-10	18					
21	0.095	-10	-17	-7	-8	-9	1	3	1	19					
22	0.1	40	30	-24	39	46	44	42	30	20					
23	0.105	-10	-12	-8	-16	-20	-5	0	-5	21					

**Figura 4-2:** Archivo CSV con datos adquiridos mediante el Myo armband.

Realizado por: Ñacato Ana, 2019

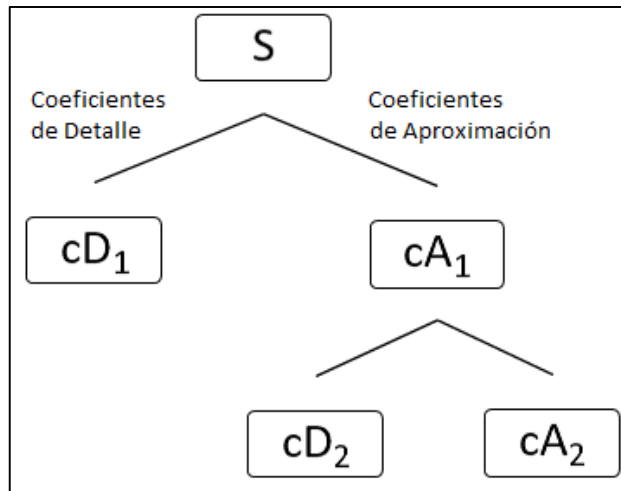
## 2.2. Sistema de procesamiento de señales

Dada la naturaleza de las señales biomédicas, en este caso señales EMG que poseen propiedades de no-linealidad, complejas, son no-estacionarias, es decir, no tienen un periodo de repetición, presentan gran variación en los datos, hace difícil analizarlos y clasificarlos.

### 2.2.1. Aplicación de la Transformada Wavelet – Análisis Multirresolución

Implementamos un sistema de procesamiento y extracción de características a través de la Transformada Wavelet Discreta (DWT). En uno de los trabajos mencionados en literatura concluyen que la Wavelet Madre Daubechies 4 y Symlet 2 son adecuadas para señales EMG a un nivel de descomposición 2. Para este trabajo de investigación aplicaremos Daubechies 4 (db4).

La señal EMG (S), pasa a través de un filtro pasa-bajas y un filtro pasa-altas, así, obtenemos un subconjunto de coeficientes de aproximación cA y un subconjunto de coeficientes de detalle cD, en la figura 5-2 observamos el diagrama de descomposición para un nivel 2.

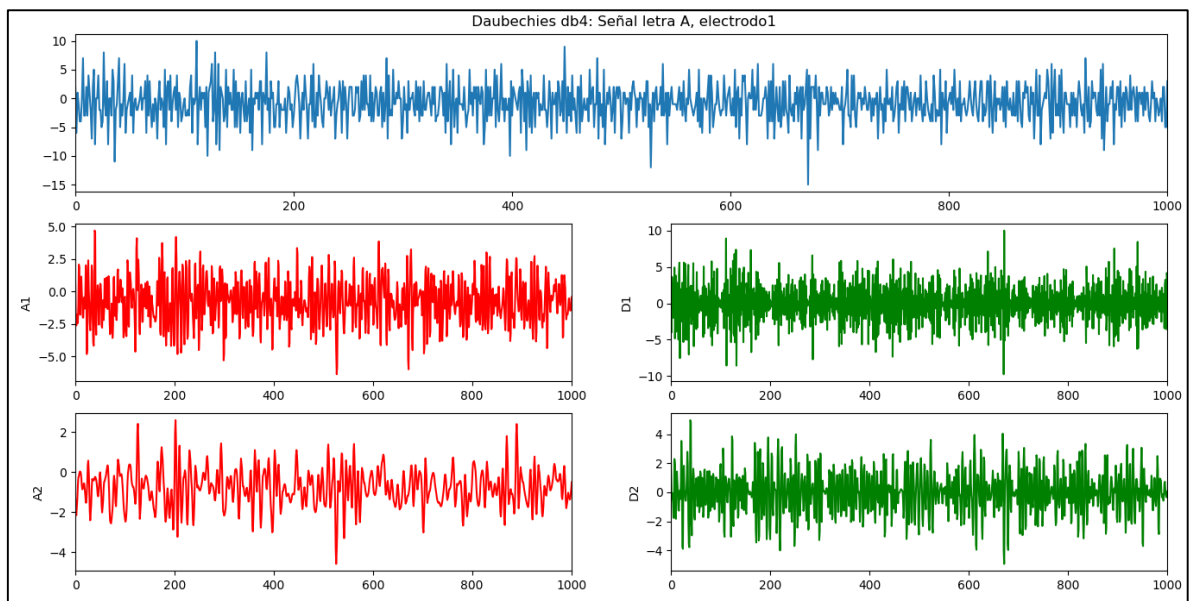


**Figura 5-2:** DWT nivel de descomposición 2

Realizado por: Ñacato Ana, 2019

Entonces, para un nivel de descomposición 2, obtenemos los subconjuntos de coeficientes wavelet  $cD_1$ ,  $cD_2$ ,  $CA_1$  y  $CA_2$ .

Estos subconjuntos de coeficientes pueden ser reconstruidos para obtener nuevamente la señal EMG, dicha reconstrucción de la señal resulta de aplicar la transformada wavelet inversa IDWT, así los subconjuntos de coeficientes reconstruidos son  $D_1$ ,  $D_2$ ,  $A_1$  y  $A_2$ . En la figura 6-2 podemos observar la gráfica de estos subconjuntos para la señal EMG del primer electrodo, realizando la señal equivalente a la letra “A”.



**Figura 6-2:** Señal  $S$  en azul,  $A_1$ ,  $A_2$  en rojo,  $D_1$ ,  $D_2$  color verde

Realizado por: Ñacato Ana, 2019

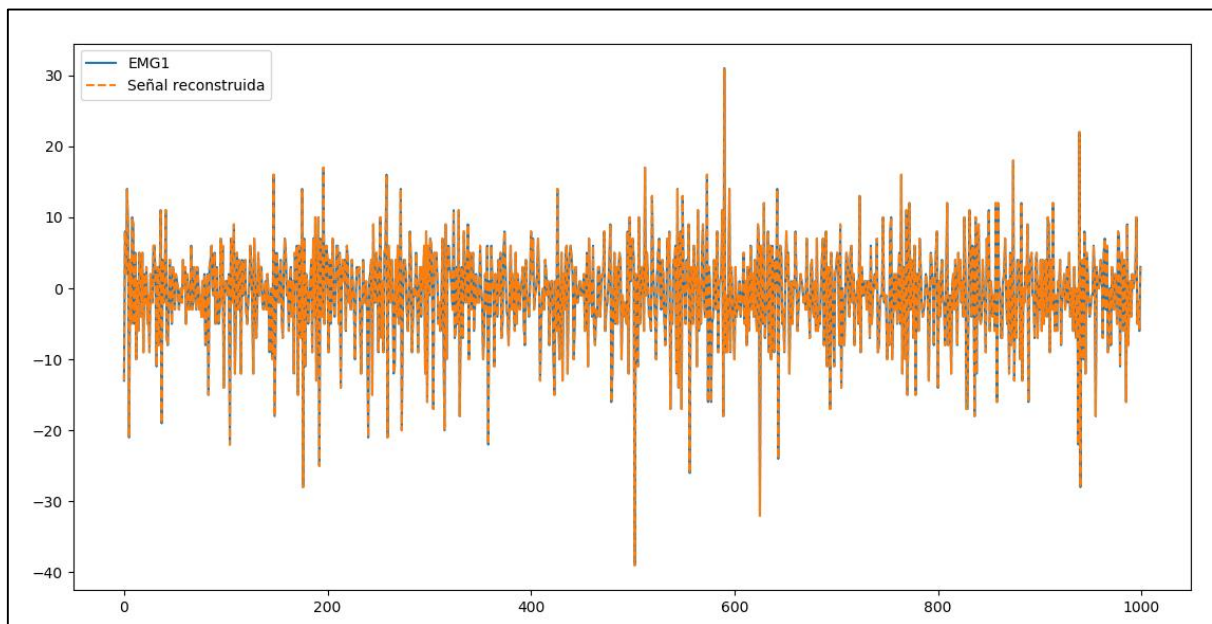
### 2.2.1.1. Reconstrucción de la señal

La reconstrucción de la señal original EMG a un nivel de descomposición 2, resulta de:

$$S = A_2 + D_2 + D_1$$

En la figura 7-2 observamos la representación gráfica de la señal del electrodo 1 para la seña A. En color Azul mostramos la señal original, mientras que en color tomate con líneas entrecortadas tenemos la señal reconstruida.

Para valorar si hay similitud entre la señal original y la señal reconstruida, se aplicó el error cuadrático medio, los valores que se obtuvieron de realizar pruebas con distintas señales las presentamos en el capítulo 3.

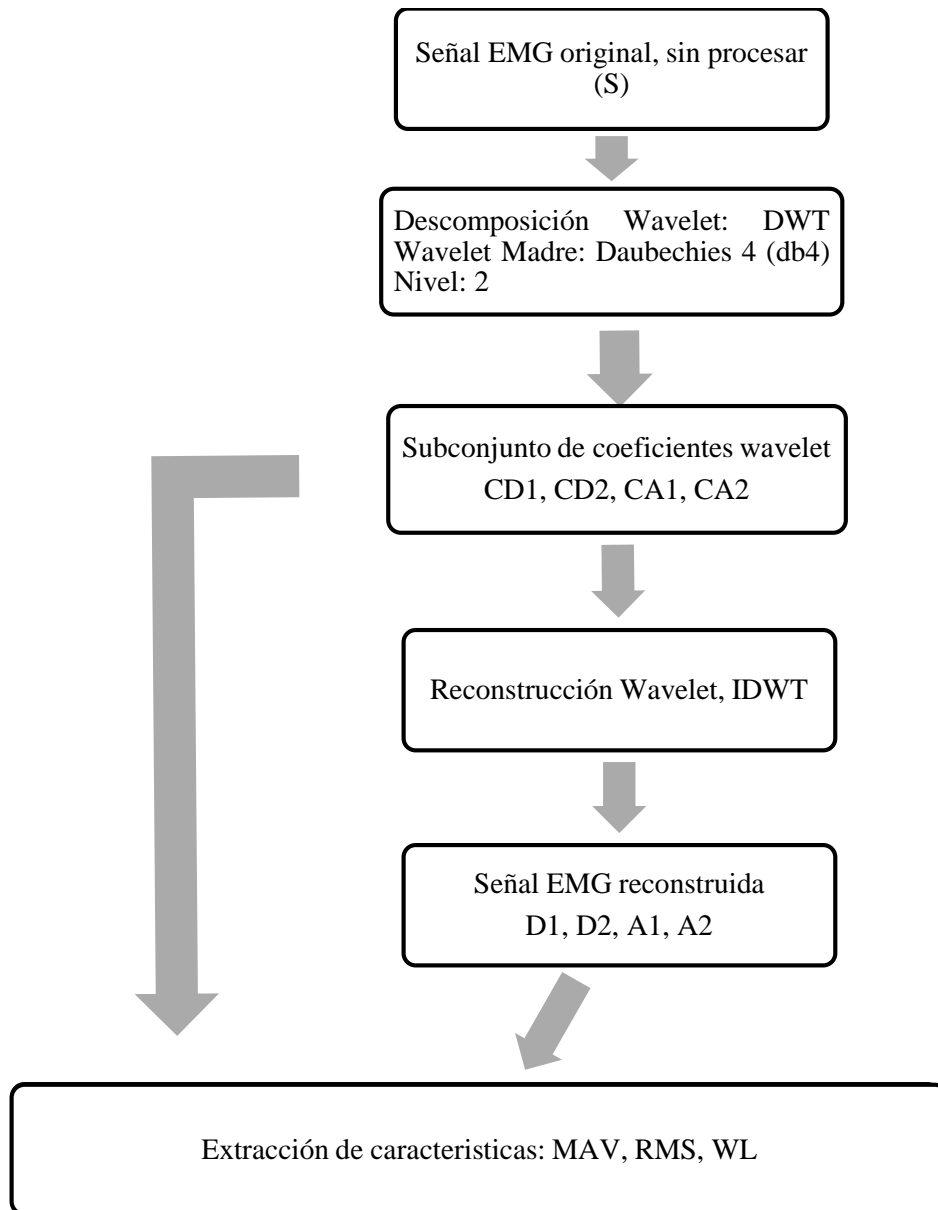


**Figura 7-2:** Reconstrucción de la señal original desde un nivel de descomposición 2

**Realizado por:** Ñacato Ana, 2019

### 2.2.2. Extracción de características

La extracción de características es un paso importante que se debe realizar, ya que brinda información útil de las señales procesadas y se reduce los datos para nuestro clasificador. En este trabajo, el valor absoluto medio (MAV), raíz cuadrática media (RMS) y Waveform length (WL) son extraídas de cada coeficiente wavelet. En la figura 8-2 mostramos el procedimiento que seguimos en esta etapa.



**Figura 8-2:** Procesamiento y extracción de características

**Realizado por:** Ñacato Ana, 2019

#### 2.2.2.1. Valor Absoluto Medio (MAV)

Es la característica más popular para el análisis de señales EMG. Es el promedio del valor absoluto de las amplitudes de la señal, está definido como:

$$MAV = \frac{1}{N} \sum_{n=1}^N |X_n| \quad (11)$$

#### 2.2.2.2. Root Mean Squared (RMS)

También es una característica popular en el análisis de señales EMG, es la media cuadrática de los datos de la señal. Matemáticamente es definido como:

$$RMS = \sqrt{\frac{1}{N} \sum_{n=1}^N (X_n)^2} \quad (12)$$

#### 2.2.2.3. Waveform Length (WL)

Para el análisis de señales EMG, la WL es definida como la longitud acumulativa de la señal EMG sobre un segmento de tiempo. La definición matemática es:

$$WL = \sum_{n=1}^N |X_{n+1} - X_n| \quad (13)$$

Donde:

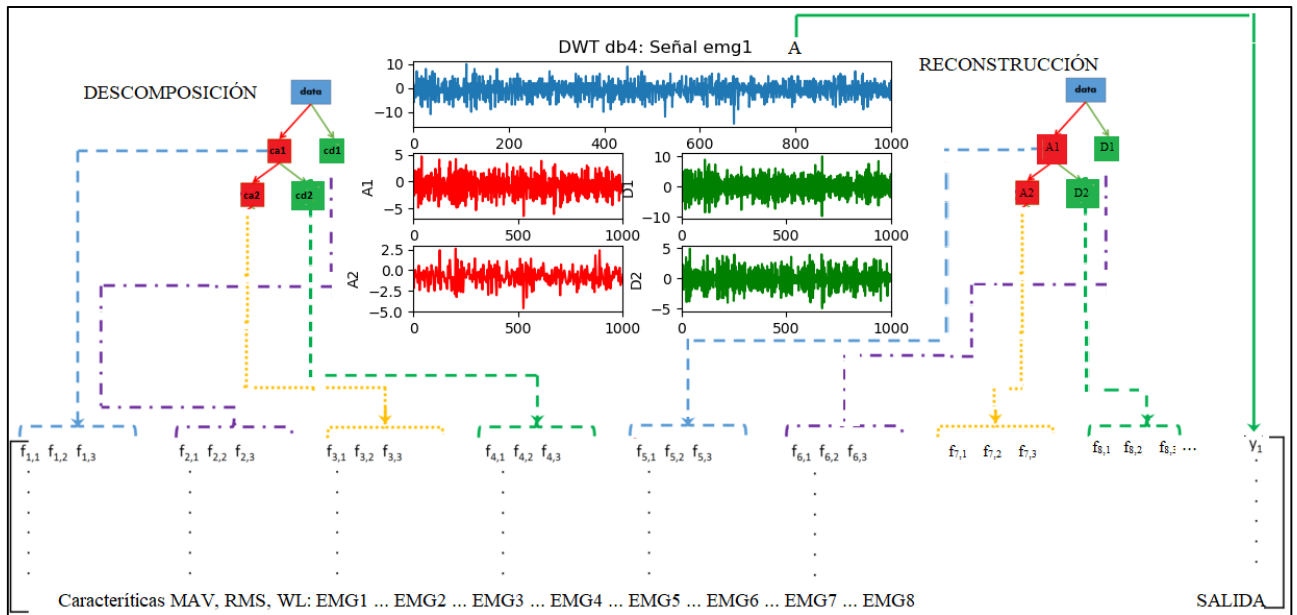
$X_n$ : Es n-ésima muestra de la señal EMG (S) o los subconjuntos de coeficientes wavelets (cD1, cD2, cA1, cA2) o las señales EMG reconstruidas (D1, D2, A1, A2)

N: La longitud de la señal EMG o los subconjuntos de datos

### 2.3. Aprendizaje automático con modelos de Clasificación

#### 2.3.1. Preparación del conjunto de datos

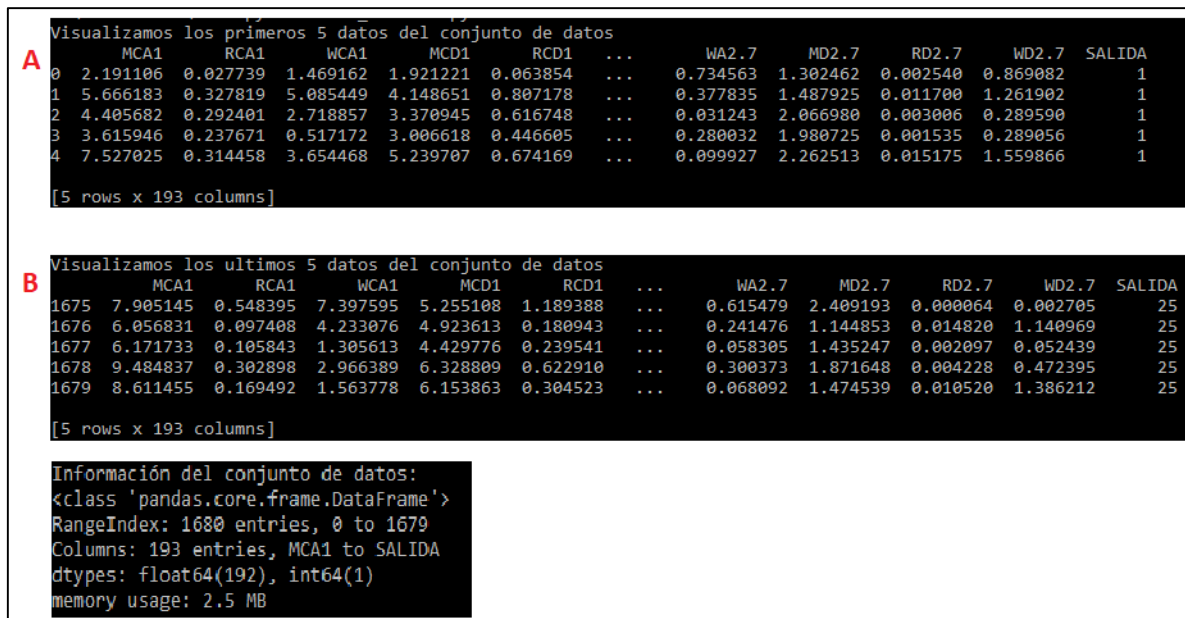
El conjunto de datos está formado por una matriz de 1680 filas x 193 columnas, las 192 columnas iniciales corresponden a las características de cada señal EMG (ver figura 9-2), mientras que la última corresponde a la Salida o etiqueta de clasificación. Para facilitar el trabajo computacional cada letra está asociada a un Número de identificación.



**Figura 9-2:** Matriz del conjunto de datos

Realizado por: Ñacato Ana, 2019

A continuación, se presenta información relevante del conjunto de datos, en la figura 10-2.A están las 5 primeras filas de los datos, en la figura 10-2.B están las últimas filas.



**Figura 10-2:** Visualización del conjunto de datos. A. Primeras 5 filas, B. Últimas 5 filas

Realizado por: Ñacato Ana, 2019



Para verificar que el conjunto de datos no tiene valores nulos o faltantes se realiza un análisis de los datos por cada columna, también se extrae 8 características estadísticas. En la figura 11-2 con la opción de “count” vemos que todas las columnas tienen 1680 datos.

Descripción del conjunto de datos:

	MCA1	RCA1	WCA1	...	RD2.7	WD2.7	SALIDA
count	1680.000000	1680.000000	1680.000000	...	1680.000000	1680.000000	1680.000000
mean	9.432384	0.505636	5.706653	...	0.021133	2.136939	13.125000
std	4.100099	0.498448	5.516138	...	0.024117	2.401418	7.335398
min	1.460733	0.000684	0.002328	...	0.000035	0.001260	1.000000
25%	6.378896	0.155505	1.662844	...	0.006169	0.611756	6.750000
50%	9.424673	0.349796	3.991677	...	0.013125	1.359393	13.500000
75%	12.690608	0.694302	8.069854	...	0.026643	2.749195	19.250000
max	22.263223	3.548876	38.456870	...	0.242688	22.241074	25.000000

[8 rows x 193 columns]

**Figura 11-2:** Información del conjunto de datos

Realizado por: Ñacato Ana, 2019

Por otro lado, en la figura 12-2 vemos la distribución de los datos en la columna de salida, observamos que cada etiqueta tiene 70 valores, es decir, hay 70 muestras por cada letra.

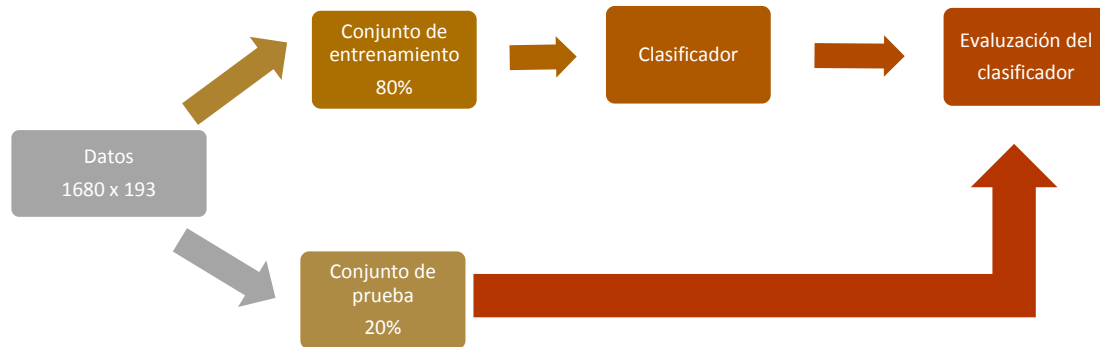
Distribución de los datos alfabeto:		gestos = {
SALIDA		"A": 1,
1 70		"B": 2,
2 70		"C": 3,
3 70		"D": 4,
4 70		"E": 5,
5 70		"F": 6,
6 70		"G": 7,
7 70		"H": 8,
8 70		"I": 9,
9 70		"K": 11,
11 70		"L": 12,
12 70		"M": 13,
13 70		"N": 14,
14 70		"O": 15,
15 70		"P": 16,
16 70		"Q": 17,
17 70		"R": 18,
18 70		"S": 19,
19 70		"T": 20,
20 70		"U": 21,
21 70		"V": 22,
22 70		"W": 23,
23 70		"X": 24,
24 70		"Y": 25,
25 70		}
dtype: int64		

**Figura 12-2:** Distribución de datos

Realizado por: Ñacato Ana, 2019

### 2.3.2. Modelos de aprendizaje

En la etapa de clasificación de señas, varios modelos de Aprendizaje supervisado son utilizados y evaluados para obtener una idea general de cuáles serían los que mejor se ajustan a nuestra investigación, tomaremos en cuenta la precisión de clasificación en cada modelo. Para ello utilizamos una de las librerías de Machine Learning, denominada Scikit-learn en Python. En la figura 13-2 observamos el procedimiento que se sigue en esta etapa.



**Figura 13-2:** Procedimiento para la clasificación de datos

Realizado por: Ñacato Ana, 2019

El primer paso a realizar es la asignación de los datos en variables que alojan las características de las señas y la salida, “X”, “y” respectivamente. Con estas variables procedemos a dividir el conjunto total, en un conjunto de entrenamiento y prueba, a este último le asignamos el 20 por ciento de los datos totales, el comando utilizado es “train\_test\_split” de la librería.

Así, obtenemos que el conjunto de entrenamiento tiene 1334 datos y el conjunto de prueba 336 datos. En la figura 14-2 mostramos el código ejecutado para este proceso.

```
1 #Separo todos los datos con las características y las etiquetas o resultados
2 X = np.array(signals.drop(['SALIDA'], 1))
3 y = np.array(signals['SALIDA'])
4
5 #Separo los datos de "train" en entrenamiento y prueba
6 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=4)
7 print('Son {} datos para entrenamiento y {} datos para prueba'.format(X_train.shape[0], X_test.shape[0]))
8
```

Son 1344 datos para entrenamiento y 336 datos para prueba

**Figura 14-2:** División de los datos

Realizado por: Ñacato Ana, 2019

Con estos datos ya separados, se procede a la clasificación con varios algoritmos, en los que se evalúa precisión de entrenamiento, precisión de prueba y tiempo de entrenamiento.

- Train score o precisión de entrenamiento, es la evaluación del modelo con los mismos datos de entrenamiento.
- Test score o precisión datos de prueba, es la evaluación del modelo con los datos de prueba.
- Time train o tiempo de entrenamiento, es el tiempo en segundos (s) ocupado por cada modelo en el entrenamiento.

Los modelos presentes en la tabla 1-2 se configuraron con parámetros generales, para decidir con que modelos seguiremos trabajando. Este entrenamiento lo realizamos tanto en un computador portátil y la plataforma embebida.

**Tabla 1-2:** Evaluación del conjunto de datos con varios clasificadores

<b>Clasificador</b>	<b>Entrenamiento Laptop</b>			<b>Entrenamiento Plataforma embebida</b>		
	<b>train_score</b>	<b>test_score</b>	<b>train_time</b>	<b>train_score</b>	<b>test_score</b>	<b>train_time</b>
<b>Random Forest</b>	1	0.943452	13.944952	1	0.943452	142.601421
<b>Gradient Boosting Classifier</b>	1	0.925595	75.849882	1	0.910714	1241.0106
<b>Naive Bayes</b>	0.920387	0.922619	0.009544	0.920387	0.922619	0.124478
<b>Logistic Regression</b>	0.994792	0.892857	3.603652	0.994792	0.892857	111.664128
<b>Red Neuronal MLP</b>	1	0.863095	3.41345	1	0.866071	111.668382
<b>Árbol de decisión</b>	1	0.842262	0.277916	1	0.845238	2.841236
<b>Vecinos más cercanos</b>	0.714286	0.553571	0.012299	0.714286	0.553571	0.19884
<b>Linear SVM</b>	1	0.059524	0.995026	1	0.059524	16.065371
<b>Promedio</b>	0.95368312	0.7503719	12.2633401	0.95368313	0.7492558	203.271807

Realizado por: Ñacato Ana, 2019

#### 2.3.2.1. Análisis de los datos obtenidos:

- El tiempo de entrenamiento en la plataforma embebida es superior en comparación al tiempo de la laptop, esto es razonable dadas las características de los equipos.

- El Gradient Boosting Classifier es el que más tiempo de entrenamiento ha requerido, seguido por Random Forest, a la vez son los que mejor precisión obtienen.
- A pesar de requerir menos tiempo de entrenamiento Vecinos más cercanos y Linear SVM son los que menor precisión han obtenido.
- Naive Bayes y Logistic Regression obtienen un porcentaje de clasificación entre el 92% y el 89% con los datos de prueba, el tiempo de entrenamiento también es menor, pero hay algo especial con estos dos modelos, el porcentaje de clasificación con los datos de entrenamiento obtenemos que es menor al 100% a pesar de evaluarlos con los mismos datos.
- Los modelos restantes tienen el 100% de precisión con los datos de entrenamiento, y un rango aceptable de precisión con los datos de prueba.

Con el análisis anterior, se determinó que Random Forest, Árbol de decisión, Red Neuronal MLP son los modelos con los que se sigue trabajando y ajustando sus parámetros para mejorar los resultados.

#### 2.3.2.2. Red Neuronal Multi-Layer Perceptron

Para este clasificador definimos los parámetros según lo establecido en la tabla 2-2. Estos valores los determinamos haciendo varias pruebas, especialmente en el número de capas ocultas y el número de neuronas para cada capa. Los resultados obtenidos de las variaciones los presentamos en el capítulo siguiente.

**Tabla 2-2:** Configuración de parámetros para `sk.neural_network.MLPClassifier`

PARAMETRO	DETALLE	SKLEARN
Número de neuronas capa de entrada	Número de características del conjunto de datos, en este caso 192.	-----
Número de capas ocultas y el número de neuronas	Definición del número de capas ocultas, con su respectiva cantidad de neuronas por capa.	<code>nn = [n,n,n]</code> <code>hidden_layer_sizes=tuple(nn[1:])</code>
Número de neuronas de salida	Equivale al número de señas a clasificar, por lo tanto, 24 neuronas.	-----

Función de activación:	Función sigmoid.	activation = 'logistic' 'logistic', the logistic sigmoid function, returns $f(x) = \frac{1}{1+e^{-x}}$
Tasa de aprendizaje:	0.01	lr = 0.01 learning_rate_init=lr
Número máximo de iteraciones	1000	max_iter=1000,

Realizado por: Ñacato Ana, 2019

### 2.3.2.3. Árbol de Decisión

En la tabla 3-2 se presenta los parámetros establecidos para este clasificador. De la misma forma se realizó varias pruebas para determinar los valores, en este caso, con al parámetro profundidad del árbol (max\_depth) hacemos varias pruebas para obtener los mejores resultados de clasificación.

**Tabla 3-2:** Configuración de parámetros para sk.DecisionTreeClassifier

PARAMETRO	DETALLE	SKLEARN
Función para medir la calidad de la división de los arboles	Medir las predicciones logradas y valorarlas para comparar de entre todas y obtener la mejor.	criterion: 'entropy'  gini: para entradas con valores continuos. entropy :para entradas categóricas
min_samples_split	Cantidad mínima de muestras que debe tener un nodo para poder subdividir.	min_samples_split=20
min_samples_leaf	Número mínimo de muestras requerido para ser un nodo de resultados	min_samples_leaf = 5
class_weight	Compensa los desbalances en la cantidad de datos	class_weight = {diccionario}

max_depth	La máxima profundidad del árbol, este parámetro se lo encuentra con validación cruzada, hasta obtener un buen porcentaje de clasificación	max_depth= n
-----------	---	--------------

Realizado por: Ñacato Ana, 2019

#### 2.3.2.4. Random Forest

En la tabla 4-2 se presenta los parámetros establecidos para Random forest clasifier. De la misma forma se realizó varias pruebas para determinar los valores, en este caso al parámetro número de árboles del bosque (n\_estimators) hacemos varias pruebas para obtener los mejores resultados de clasificación.

**Tabla 4-2:** Parámetros para sk.RandomForestClassifier

PARAMETRO	DETALLE	SKLEARN
Función para medir la calidad de la división de los arboles	Medir las predicciones logradas y valorarlas para comparar de entre todas y obtener la mejor.	criterion: 'entropy' gini: para entradas con valores continuos. entropy :para entradas categóricas
n_estimators	Cantidad de árboles en el bosque	n_estimators = n
class_weight	Compensa los desbalances en la cantidad de datos	class_weight = {diccionario}
max_depth	La máxima profundidad del árbol, este parámetro se lo encuentra con validación cruzada, hasta obtener un buen porcentaje de clasificación	max_depth= 8

Realizado por: Ñacato Ana, 2019

## 2.4. Diseño de la interfaz

El diseño de la interfaz de traducción se lo realizó a través de la librería de Python, llamada PyQt4, Esta interfaz permite adquirir los datos de las señales puras de los electrodos y guardarlos en un archivo CSV, para el posterior procesamiento en la FPGA. También en el cuadro de texto muestra el resultado de la clasificación.

En la figura 15-2, mostramos la interfaz que se ejecuta en una laptop.



**Figura 15-2:** Interfaz de traducción

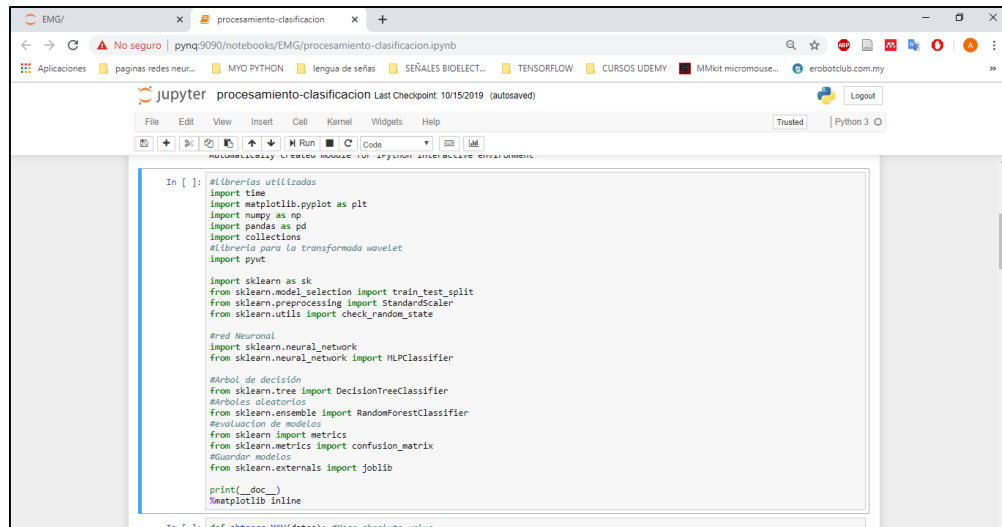
Realizado por: Ñacato Ana, 2019

En donde:

1. Botón de Inicio, para dar comienzo a la traducción
2. Botón Guardar archivo, nos permite guardar los datos de los sensores
3. Botón Stop, finalización del programa
4. Espacio donde se muestra la seña traducida
5. Botón para una nueva traducción

## 2.5. Implementación sobre la FPGA

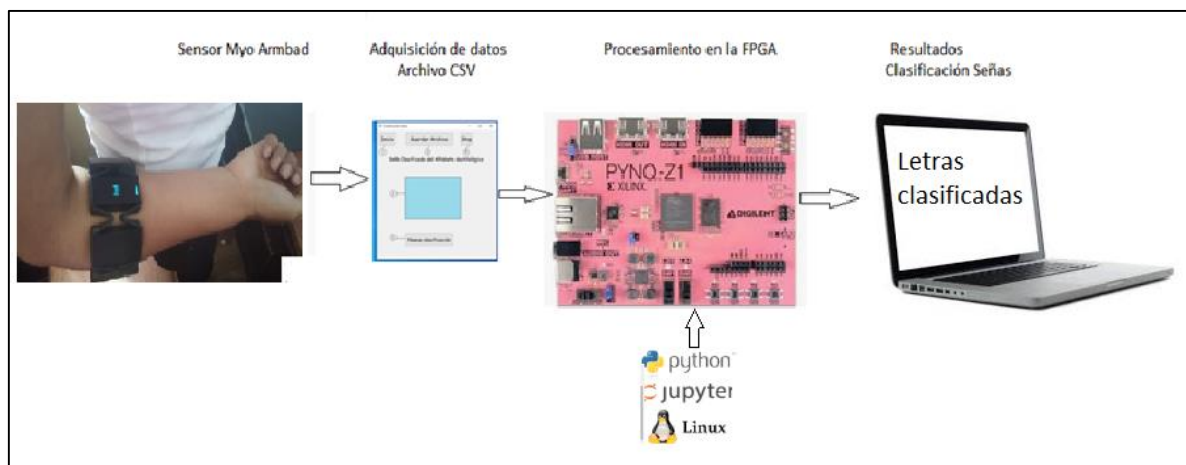
Una vez, que el archivo se ha creado con los datos del sensor, el programa de la FPGA desarrollado sobre Jupyter Notebook se ejecuta, procesa las señales, extrae las características y con este conjunto de datos clasifica la señal y muestra la letra a la que pertenece. En la figura 16-2 mostramos parte del código y las librerías utilizadas.



**Figura 16-2:** Librerías utilizadas a través de Jupyter Notebook de la FPGA

Realizado por: Ñacato Ana, 2019

### 2.5.1. Diagrama de bloques de la interfaz de traducción del alfabeto dactilológico

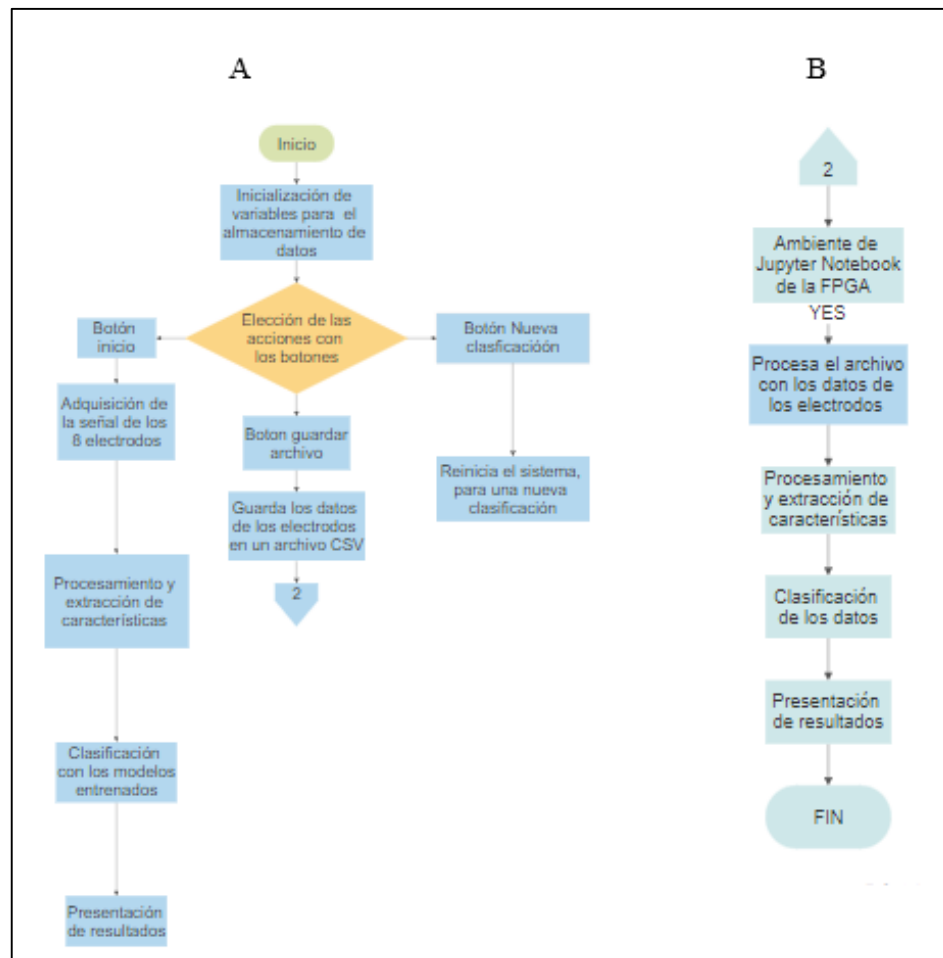


**Figura 17-2:** Diagrama de bloques de la interfaz de traducción del alfabeto dactilológico

Realizado por: Ñacato Ana, 2019



### 2.5.2. Diagrama de flujo del funcionamiento de la interfaz de traducción del alfabeto dactilológico



**Figura 18-2:** Diagrama de flujo, A. Interfaz de traducción, B. Procesamiento de la FPGA

Realizado por: Ñacato Ana, 2019

El centro educativo denominado “Unidad Educativa Especializada Sordos de Chimborazo” de la ciudad de Riobamba trabaja con personas con discapacidad auditiva, con el cual, se intentó establecer un acuerdo para que se nos preste las facilidades de realizar algunas pruebas de aceptabilidad del dispositivo, pero su respuesta fue negativa. Por lo tanto, no se pudieron realizar dichas pruebas.

## CAPÍTULO III

### 3. MARCO DE RESULTADOS

En este capítulo evaluamos los resultados sobre la plataforma embebida FPGA, está dividida en dos partes, la primera, abarca el procesamiento de señales, específicamente la reconstrucción de la señal EMG con la Transformada Wavelet Discreta Inversa, la métrica aplicada para valorar la similitud entre la señal original y la señal reconstruida es el Error Cuadrático Medio. Los resultados al aplicar esta métrica deben ser menores a 1 y cercanos a 0, mientras los valores más cerca estén de 0, la similitud entre las dos señales es alta y no hemos perdido información. El Error Cuadrático Medio, está definido como:

$$\bar{\varepsilon}^2 = \frac{1}{N} \sum_{n=1}^N |x(n) - \tilde{x}(n)|^2 \quad (1)$$

Donde:  $x(n)$ , es la señal original

$\tilde{x}(n)$ , es la señal reconstruida

N, número de muestras (N = 1000)

En la segunda parte, determinamos el valor de los parámetros de los modelos de aprendizaje. Una vez encontrado estos valores, evaluamos los modelos con diferentes métricas. Para estas pruebas, contamos con 336 muestras, distribuidas para cada letra de acuerdo a la tabla 1-3.

**Tabla 1-3:** Datos de prueba

Salida	Etiqueta	Cantidad / seña
A	1	18
B	2	17
C	3	12
D	4	13
E	5	16
F	6	16

G	7	14
H	8	15
I	9	11
K	11	18
L	12	17
M	13	7
N	14	17
O	15	13
P	16	10
Q	17	14
R	18	15
S	19	17
T	20	17
U	21	7
V	22	9
W	23	16
X	24	12
Y	25	15
		336

Realizado por: Ñacato Ana, 2019

### 3.1. Reconstrucción de señales

La señal reconstruida a un nivel de descomposición igual a 2 es:  $S = A_2 + D_2 + D_1$

Como disponemos de 8 electrodos para la adquisición de señales EMG, para esta prueba únicamente tomamos en cuenta la señal adquirida por el electrodo 1, en cada una de las 24 letras. Realizamos la evaluación del error cuadrático medio para 20 señales originales con su respectiva señal reconstruida. Ver Anexo A.

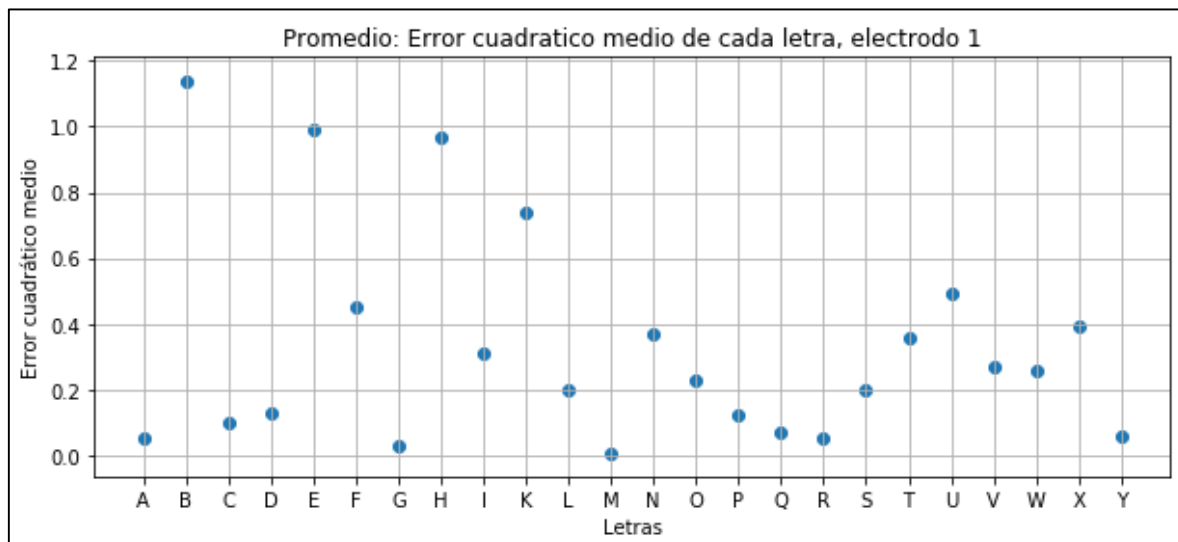
El promedio de los resultados obtenidos están en la tabla 2-3.

**Tabla 2-3:** Promedio: Error cuadrático medio  
para cada una de las letras

Letra	Error cuadrático medio
A	0.05
B	1.14
C	0.1
D	0.13
E	0.99
F	0.45
G	0.03
H	0.97
I	0.31
K	0.74
L	0.2
M	0.008
N	0.37
O	0.23
P	0.12
Q	0.07
R	0.05
S	0.2
T	0.36
U	0.49
V	0.27
W	0.26
X	0.39
Y	0.06

Realizado por: Ñacato Ana, 2019

La representación gráfica de los datos anteriores están en el gráfico 1-3.



**Gráfico 1-3:** Error cuadrático medio del electrodo 1 para cada una de las letras

Realizado por: Ñacato Ana, 2019

Del gráfico anterior, observamos que el promedio del Error cuadrático medio en la mayoría de las letras es  $< 1$ , específicamente, para 20 letras está en el rango de 0 y 0.5, lo que significa que hay una alta similitud entre la señal original y la señal reconstruida.

Únicamente, el error cuadrático medio de la letra B, es superior a 1, esto no significa que la señal reconstruida es totalmente diferente a la original, más bien que hay una pérdida de información, pero no afecta en la extracción de características.

### 3.2. Clasificación de señas

Para determinar los parámetros base y evaluar los modelos de clasificación de las señas, utilizamos 336 muestras que están detalladas en la tabla 1-3.

#### 3.2.1. Elección de parámetros

##### 3.2.1.1. Multi Layer Perceptron

La determinación de parámetros se inicia en un modelo sencillo de Multi Layer Perceptron con 2 capas ocultas, después con 3 y 4 capas ocultas, el número de neuronas por cada capa también variamos desde 10 hasta 410 neuronas.

**Tabla 3-3:** Porcentaje de clasificación correctamente para varios modelos MLP

	2 Capas ocultas [n,n]		3 Capas ocultas [n,n,130]		4 Capas ocultas [n,n,390,170]	
N° neuronas (n)	Train score	Test score	Train score	Test score	Train score	Test score
10	0.842262	0.77381	0.761905	0.67559	0.042411	0.038690
30	0.977679	0.880952	0.965774	0.84226	0.044643	0.029762
50	0.99256	0.901786	0.998512	0.875	0.046875	0.020833
70	0.999256	0.910714	0.999256	0.89285	0.081845	0.044643
90	1	0.907738	0.999256	0.86904	0.046875	0.026786
110	1	0.907738	1	0.89881	1	0.839286
130	1	0.904762	1	0.88392	1	0.809524
150	1	0.895833	1	0.89583	1	0.824405
170	1	0.91369	1	0.90178	1	0.824405
190	1	0.892857	1	0.89285	1	0.830357
210	1	0.907738	1	0.88988	1	0.815476
230	1	0.895833	1	0.87202	1	0.842262
250	1	0.892857	1	0.90476	1	0.833333
270	1	0.889881	1	0.90476	1	0.824405
290	1	0.895833	1	0.89583	1	0.833333
310	1	0.895833	1	0.90476	1	0.827381
330	1	0.901786	1	0.89881	1	0.815476
350	1	0.901786	1	0.8869	1	0.833333
370	1	0.907738	1	0.90178	1	0.830357
390	1	0.895833	1	0.91071	1	0.800595
410	1	0.895833	1	0.88392	1	0.842262

Realizado por: Ñacato Ana, 2019

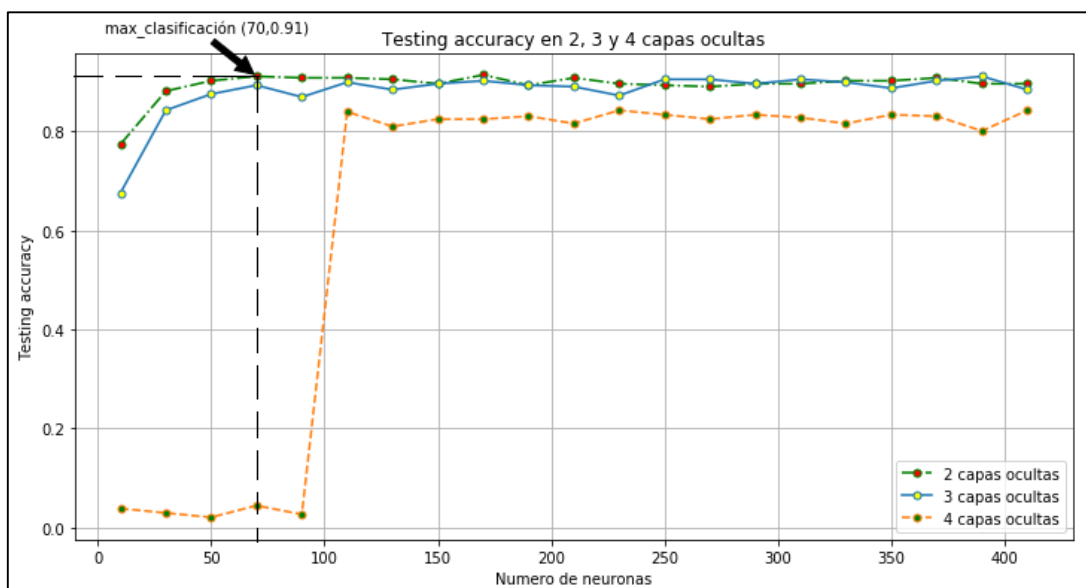
Los datos de la tabla 3-3 están representados el gráfico 2-3, donde vemos la relación entre el número de neuronas vs porcentaje de clasificación correctamente con los datos de prueba. La línea verde

representa a una red neuronal MLP con 2 capas ocultas, línea azul representa a una red neuronal MLP con 3 capas ocultas, la línea tomate entrecortada es una red neuronal con 4 capas ocultas.

Del gráfico, observamos que el modelo con 2 capas ocultas es el que mejor porcentaje de clasificación alcanza, con 30 neuronas en cada capa, obtiene el 88% de datos bien clasificados, a partir de esta cantidad de neuronas el modelo responde a un porcentaje de clasificación casi constante entre el 89% y 91% de datos bien clasificados, siendo el último porcentaje correspondiente a 70 neuronas.

Para el modelo con 3 capas ocultas el porcentaje de clasificación varía entre el 84% y el 91%, con 390 neuronas en cada capa el modelo logra el 91% de datos bien clasificados.

Finalmente, con 4 capas ocultas el porcentaje de clasificación es el más bajo, con 10 neuronas por capa alcanza el 3% de clasificación, el porcentaje máximo de datos bien clasificados es del 84% porcentaje mínimo ya alcanzado con 3 capas ocultas y 30 neuronas.



**Gráfico 2-3:** Gráfica del conjunto de datos evaluados con 2, 3 y 4 capas ocultas

**Realizado por:** Ñacato Ana, 2019

Con lo expuesto anteriormente, podemos decir que el modelo de red neuronal Multi Layer Perceptron alcanza los mejores resultados de datos bien clasificados, con 2 y 3 capas ocultas. Sin embargo, el modelo con 3 capas ocultas requiere más recursos y el costo computacional es alto. Por lo tanto, el modelo de red neuronal lo definimos con 2 capas ocultas y cada una con 70 neuronas, obteniendo el 91% de los datos de prueba bien clasificados.

### 3.2.1.2. Árbol de decisión

Para este modelo árbol de decisión, creamos diferentes modelos asignando un valor de max\_depth desde 1 hasta 10. La evaluación de este parámetro lo realizamos con validación cruzada = 10, los resultados obtenidos están en la tabla 4-3.

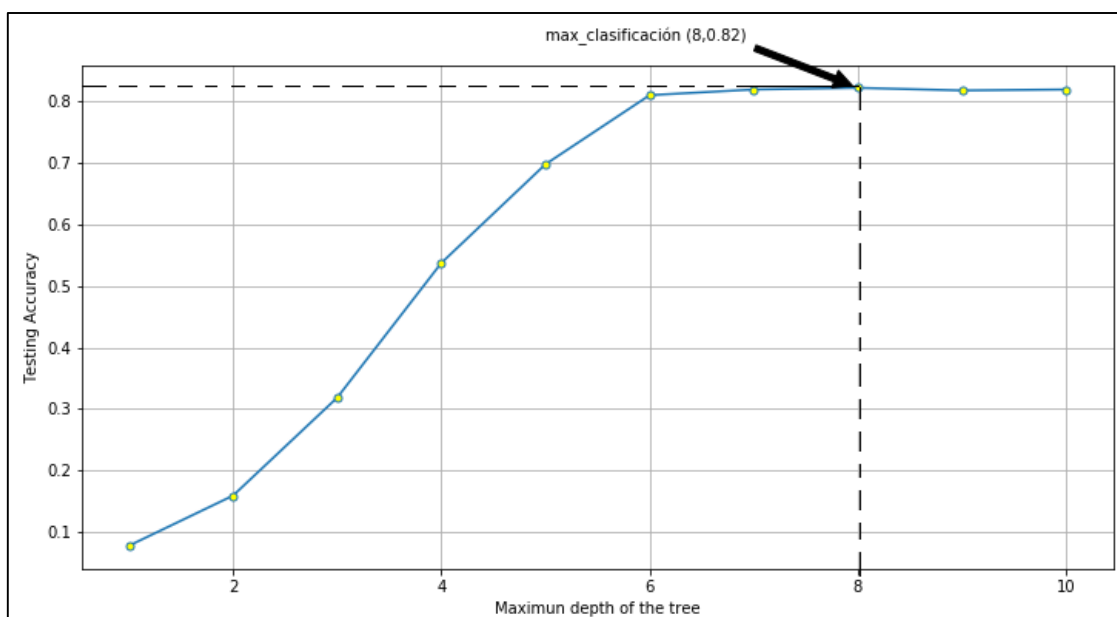
**Tabla 4-3:** Porcentaje de clasificación para el modelo Árbol de decisión, variando la profundidad máxima

Árbol de decisión	
max_depth	Average Accuracy
1	0.077413
2	0.158975
3	0.318367
4	0.537528
5	0.698141
6	0.81057
7	0.819805
8	0.822484
9	0.818232
10	0.819832

Realizado por: Ñacato Ana, 2019

El porcentaje de datos bien clasificados para el modelo Árbol de decisión variando la profundidad máxima del arbol, empieza en el 7% y alcanza un máximo del 82%. A partir, de max\_depth = 6 el porcentaje de clasificación se mantiene constante en el 81%, solo con max\_depth = 8 vemos un incremento al 82%, siendo este, el valor del parámetro elegido para el modelo. En el gráfico 3-3 observamos a detalle dicho comportamiento.





**Gráfico 3-3:** Relación entre la variación de la profundidad máxima del árbol vs porcentaje de clasificación.

Realizado por: Ñacato Ana, 2019

### 3.2.1.3. Random Forest

Al ser este modelo una variación del modelo Árbol de decisión, definimos el parámetro `max_depth = 8`, el parámetro que deseamos encontrar es el número de árboles del bosque, para ello creamos varios modelos, asignando valores comprendidos entre 10 y 960. Los resultados de datos bien clasificados están en la tabla 5-3.

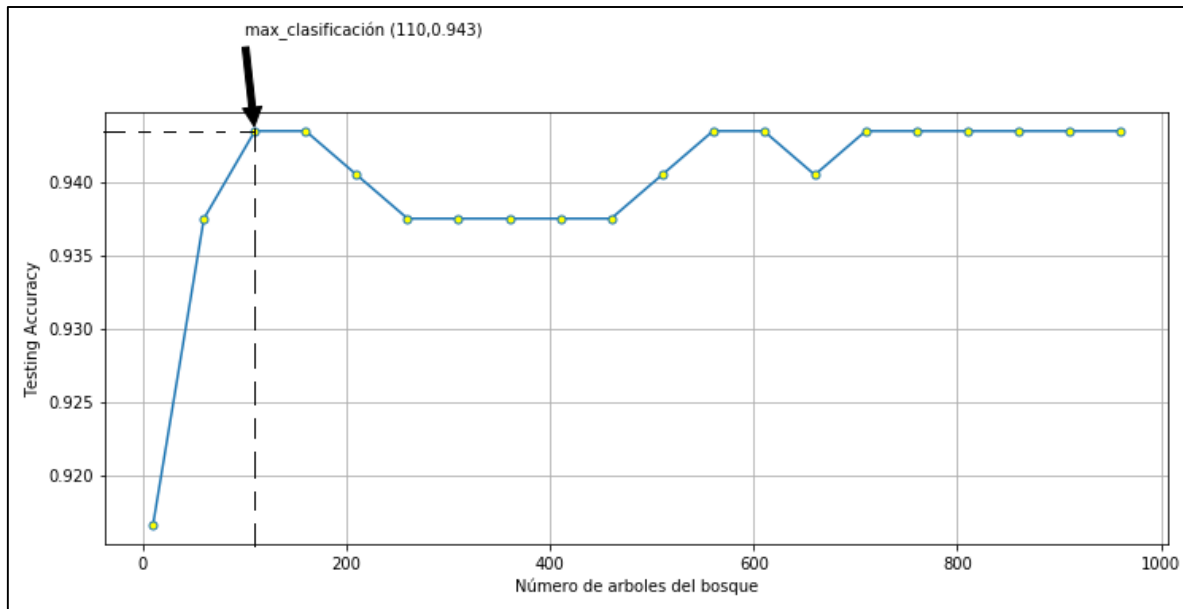
**Tabla 5-3:** Porcentaje de datos bien clasificados para el modelo Random Forest, variando el número de árboles.

Random Forest	
Número de arboles	Average Accuracy
10	0.916667
60	0.9375
110	0.943452
160	0.943452
210	0.940476
260	0.9375

310	0.9375
360	0.9375
410	0.9375
460	0.9375
510	0.940476
560	0.943452
610	0.943452
660	0.940476
710	0.943452
760	0.943452
810	0.943452
860	0.943452
910	0.943452
960	0.943452

**Realizado por:** Ñacato Ana, 2019

En este modelo, el porcentaje de datos bien clasificados alcanzó un rango del 91% al 94%. Con 110 a 210 árboles del bosque se obtuvo el 94%. A partir de los 510 árboles del bosque, se obtuvo un porcentaje de clasificación casi constante del 94%. En el gráfico 4-3 visualizamos estos resultados.



**Gráfico 4-3:** Relación entre la variación del número de árboles del bosque vs porcentaje de clasificación.

Realizado por: Ñacato Ana, 2019

El resultado de realizar varias pruebas a los modelos, los parámetros los definimos de acuerdo a la siguiente tabla.

**Tabla 6-3:** Parámetros de los modelos y su porcentaje de clasificación

MODELO	PARÁMETRO	SKLEARN	Porcentaje de datos bien clasificados
MLP	2 capas ocultas 70 neuronas por cada capa	nn = [70,70]	91%
ARBOL DE DECISIÓN	Profundidad máxima del árbol.	max_depth = 8	82%
RANDOM FOREST	Profundidad máxima del árbol. Número de árboles del bosque	max_depth = 8 n_estimators = 110	94%

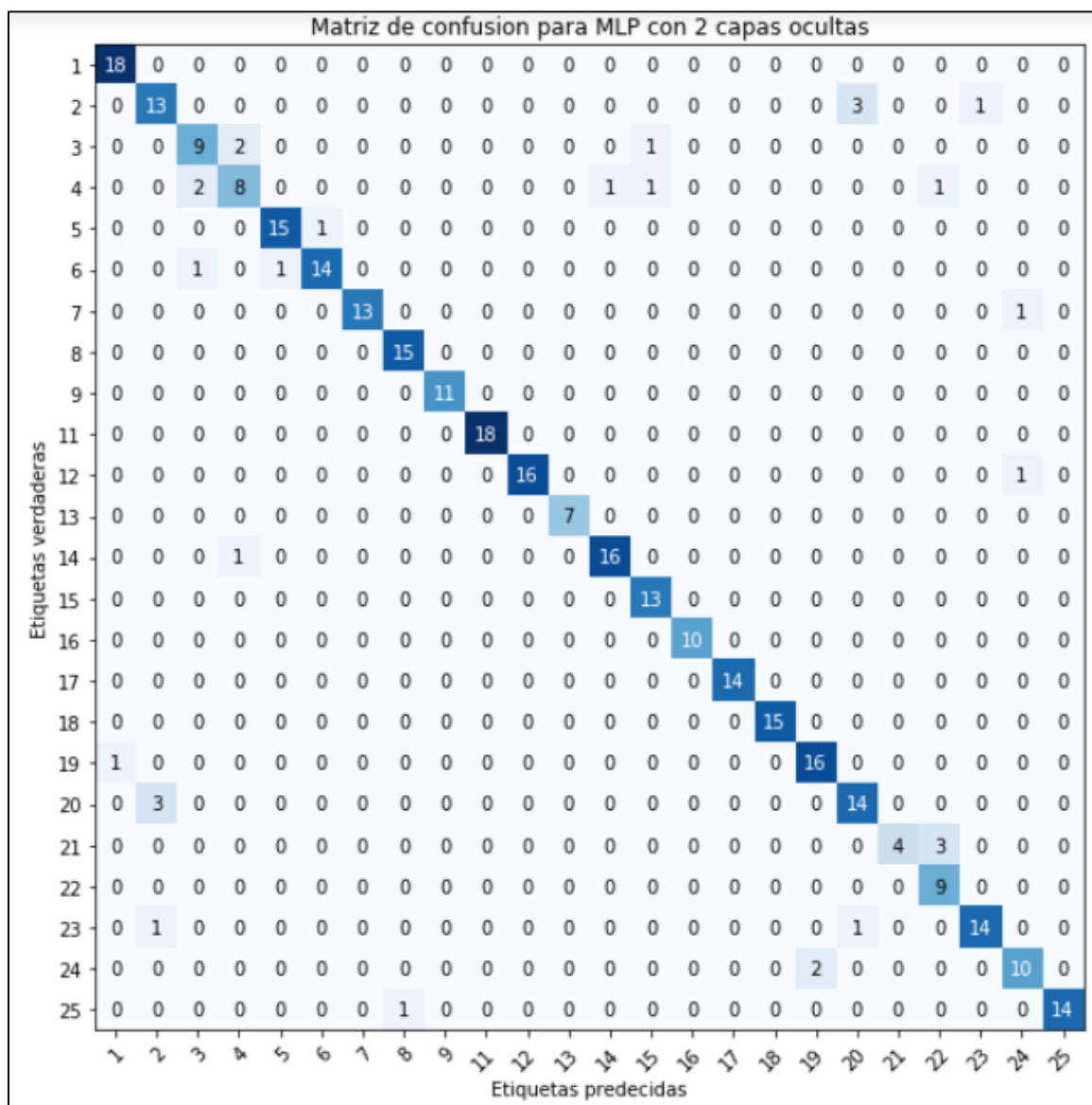
Realizado por: Ñacato Ana, 2019

### 3.2.2. Pruebas de datos correctamente clasificados

Para estas pruebas, a través de la matriz de confusión de cada uno de los modelos se visualiza el número de letras clasificadas correctamente. Las filas representan a las Salidas o etiquetas verdaderas

y en las columnas están las Salidas o etiquetas predichas para cada letra. La diagonal principal representa el valor de muestras que son clasificadas correctamente.

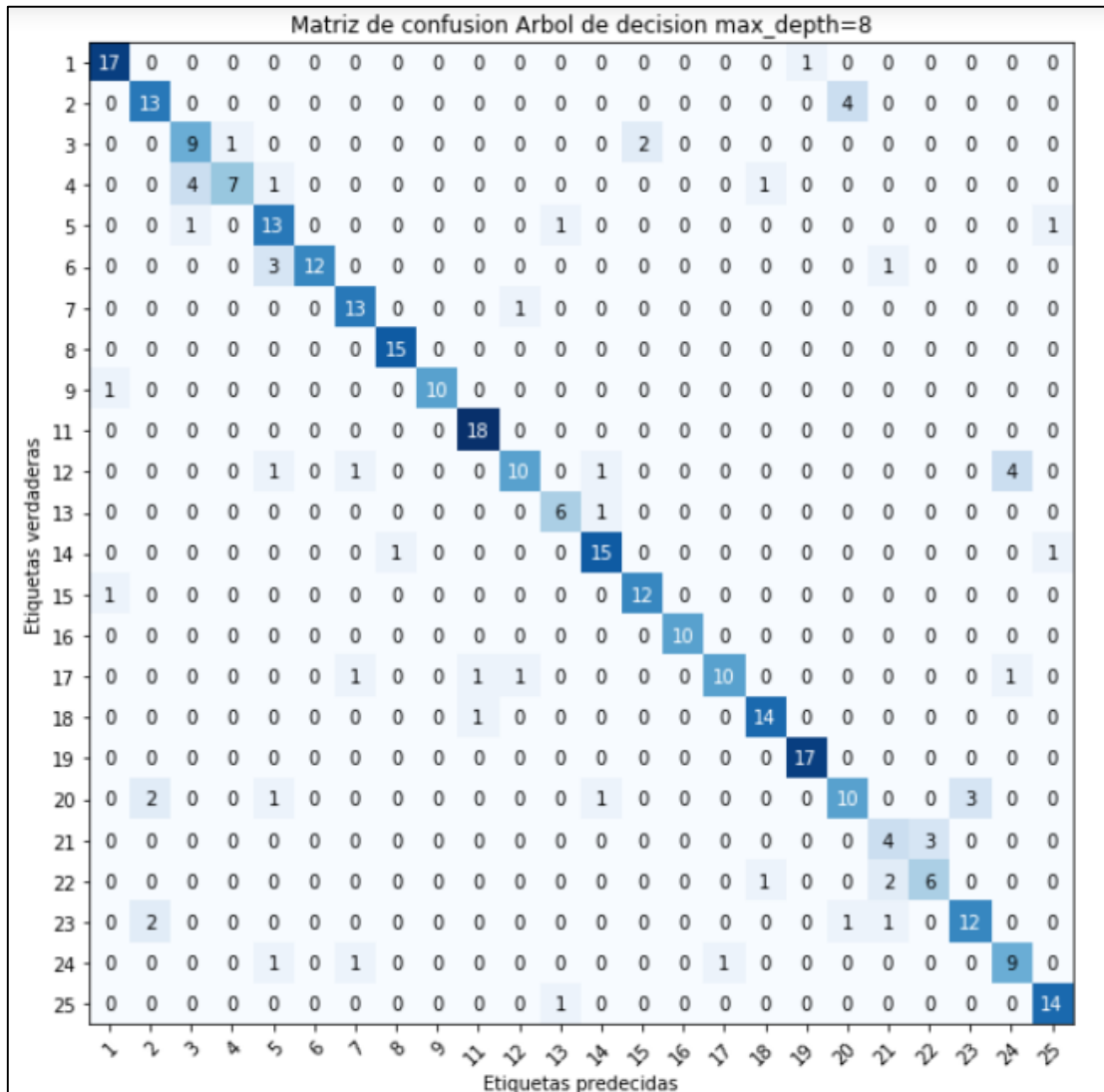
### Matriz de confusión del modelo Multi Layer Preceptron



**Figura 1-3:** Matriz de confusión del modelo MLP

Realizado por: Ñacato Ana, 2019

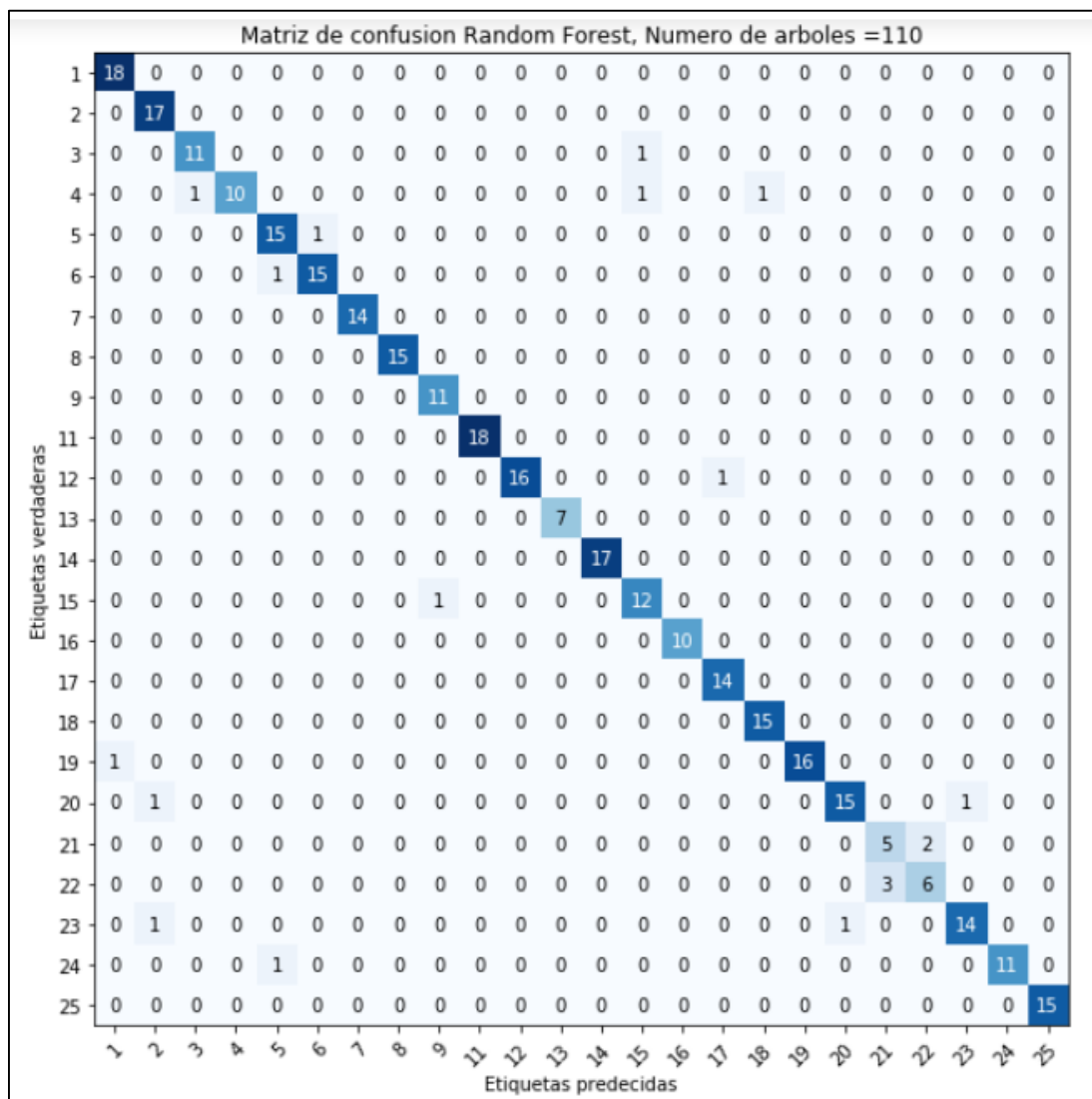
## Matriz de confusión del modelo Árbol de decisión



**Figura 2-3:** Matriz de confusión del modelo Árbol de decisión

Realizado por: Ñacato Ana, 2019

### Matriz de confusión del modelo Random Forest



**Figura 3-3:** Matriz de confusión del modelo Random Forest

Realizado por: Ñacato Ana, 2019

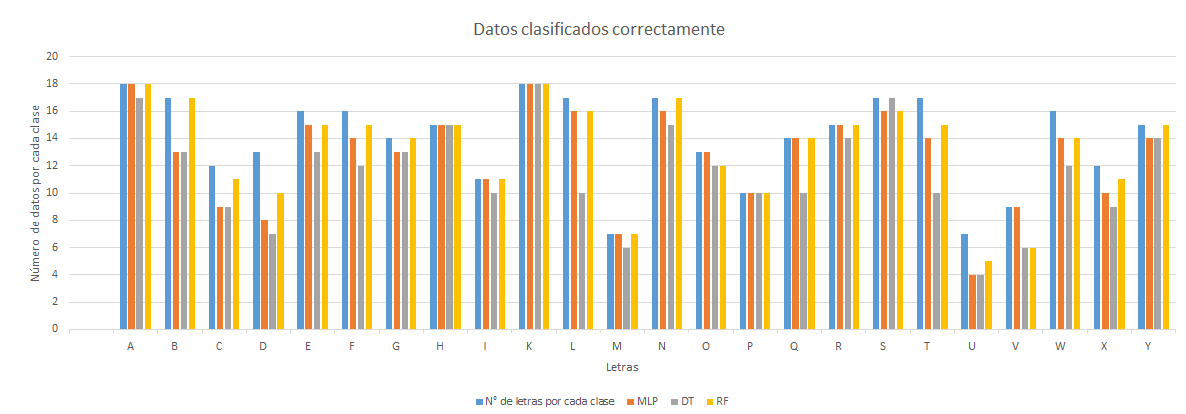
Los datos de la diagonal principal de cada una de las matrices se muestran en el gráfico 5-3, para una interpretación más precisa. En el anexo B, se presenta en forma de tabla el número de datos correctamente clasificados para cada uno de los modelos.

Al analizar estos datos se obtiene que los modelos presentan un alto número de muestras clasificadas correctamente, hay alrededor de 2 a 3 muestras que se predicen incorrectamente. El modelo con mejor número de muestras clasificadas correctamente es Random Forest.

La precisión indica el porcentaje de muestras bien clasificadas, por lo tanto, para la clase de la letra A, en dos de los tres modelos es del 100%, esto quiere decir, que las 18 muestras existentes en el conjunto de prueba fueron predichas correctamente con la clase A.

Analizando los resultados obtenidos para la clase P, tenemos que la precisión en todos los modelos es del 100%, lo que significa que las 10 muestras existentes en el conjunto de prueba para esta clase, fue clasificada correctamente y ninguna otra muestra se clasificó en esta clase.

La menor precisión lo obtenemos para la clase U en los tres modelos, ya que, de las 7 muestras presentes en el conjunto de prueba, solo 4 fueron clasificadas correctamente por los modelos MLP y árbol de decisión y 5 para el modelo Random Forest. Las métricas de precisión y recall para cada clase se muestra en el Anexo C.



**Gráfico 5-3:** Relación del número de datos verdaderos por cada clase vs datos correctamente clasificados por cada modelo.

**Realizado por:** Ñacato Ana, 2019

## CONCLUSIONES

- Se diseñó e implementó una interfaz para la traducción de caracteres del alfabeto dactilológico a palabras utilizando sensores musculares y reconocimiento de patrones.
- La metodología aplicada en este trabajo de titulación contribuyó a que se cumplan los parámetros establecidos, se ejecutó el diseño de la interfaz y se implementó el procesamiento de datos en la FPGA.
- Las señales electromiográficas son del tipo no estacionarias, por lo que se debe aplicar un procesamiento de señales tomando en cuenta esta característica.
- La Transformada Wavelet Discreta con su análisis Multirresolución se aplicó a cada uno de las señales de los electrodos, permitiendo separar la señal en sus componentes de alta y baja frecuencia y de cada subconjunto obtener información útil a través de la extracción de características para formar el conjunto de datos.
- Para determinar el algoritmo de aprendizaje supervisado más óptimo en la clasificación de las señas se hicieron variaciones en los parámetros base, como: el número de capas ocultas y el número de neuronas por cada capa en una red neuronal Multi Layer Perceptron. La profundidad máxima del árbol en el algoritmo árbol de decisión y para Random Forest el número de árboles en el bosque.
- La programación de la plataforma embebida FPGA sobre Jupyter Notebook facilitó la ejecución, procesamiento de señales y clasificación de las señas.
- Una vez implementada la interfaz se realizó pruebas de funcionamiento con un porcentaje de señas correctamente clasificadas del 94% con el algoritmo Random Forest.



## RECOMENDACIONES

- Se recomienda a la Institución universitaria establecer convenios con los centros que acogen a personas con discapacidad auditiva para facilitar la realización de pruebas de campo con equipos desarrollados como el presentado en este estudio.
- La ubicación del sensor debe estar en el antebrazo derecho cercano al codo (músculos Flexores radiales del carpo, flexores superficiales de los dedos), con el módulo que contiene el ícono del brazalete en la parte superior.
- Una vez determinadas las mejores configuraciones de las técnicas y herramientas de clasificación, se recomienda utilizarlas a través de un entrenamiento más exhaustivo para reducir el costo computacional y tiempo de respuesta.
- Se recomienda para trabajos futuros la integración en un solo sistema la adquisición de datos y presentación de resultados en la FPGA.

## BIBLIOGRAFÍA

**ABDUO, M. y GALSTER, M.** Myo Gesture Control Armband for Medical Applications. [en línea], 2015 [Consulta: 5 marzo 2018]. Disponible en: [https://ir.canterbury.ac.nz/bitstream/handle/10092/14449/hons\\_1502.pdf?sequence=1](https://ir.canterbury.ac.nz/bitstream/handle/10092/14449/hons_1502.pdf?sequence=1).

**TASPINAR, A.** *A guide for using the Wavelet Transform in Machine Learning*, [en línea], 2018. [Consulta: 12 enero 2019]. Disponible en: <http://ataspinar.com/2018/12/21/a-guide-for-using-the-wavelet-transform-in-machine-learning/>.

**ALDÁS, A. y VANEGAS, J.** Diseño y construcción de un módulo mecatrónico adaptable que permita la movilidad de sillas de ruedas, controlada por señales electroencefalográficas, para la empresa Algoritmun Cía. Ltda. [en línea], (Tesis), (Pregrado), Escuela Superior Politécnica del Ejército, Quito, Ecuador, 2016, [Consulta: 27 septiembre 2018]. Disponible en: <http://repositorio.espe.edu.ec/handle/21000/12078>.

**AUQUILLA, D., PALACIO-BAUS, K. y SAQUICELA, V.** "Reconocimiento de caracteres del alfabeto dactilológico mediante redes neuronales artificiales: un enfoque experimental". *MASKANA*, [en línea], 2015, Ecuador, [Consulta: 27 septiembre 2018]. ISSN 2477-8893. Disponible en: [https://dspace.ucuenca.edu.ec/bitstream/123456789/23793/1/2015\\_TIC.EC\\_6.pdf](https://dspace.ucuenca.edu.ec/bitstream/123456789/23793/1/2015_TIC.EC_6.pdf)

**BETANCOURT O, G., GIRALDO, E. y FRANCO, J.,.** "Reconocimiento de patrones de movimiento a partir de señales electromiográficas". *Scientia et Technica* [En línea], vol. 3, no. 26, pp. 53-58., 2004, [Consulta: 30 mayo 2018]. ISSN 0122-1701. DOI 10.22517/23447214.7045. Disponible en: <http://revistas.utp.edu.co/index.php/revistaciencia/article/view/7045>.

**BOTSCIENCIE.** *Sensor Muscular EMG (Electromiografía)*. [en línea]. 2013. [Consulta: 18 octubre 2018]. Disponible en: [http://botscience.net/store/index.php?route=product/product&manufacturer\\_id=28&product\\_id=209](http://botscience.net/store/index.php?route=product/product&manufacturer_id=28&product_id=209)

**CABALLERO, K., DUQUE, L., CEBALLOS, S., RAMÍREZ, J. y PELÁEZ, A.** "Concepto básicos para el análisis electromiográfico", *Revista CES Odontología* [en línea]. 2002, (Colombia), volumen 15, Número 1, [Consulta: 18 octubre 2018]. Disponible en: <http://revistas.ces.edu.co/index.php/odontologia/article/viewFile/596/350>.

**CAMERON, J. y SKOFRONICK, J.G.** *Medical physics*. [en línea], Michigan - EEUU, Wiley. 1978. [Consulta: 13 diciembre 2018], ISBN 0471131318. Disponible en: [https://books.google.com.ec/books/about/Medical\\_physics.html?id=xjtrAAAAMAAJ&redir\\_esc=y](https://books.google.com.ec/books/about/Medical_physics.html?id=xjtrAAAAMAAJ&redir_esc=y)

**CIFUENTES, I.**, Diseño Y Construcción De Un Sistema Para La Detección De Señales Electromiográficas (Tesis) (Pregrado), [en línea], Universidad Autónoma de Yucatán, Yucatán-México, 2010, [Consulta: 13 diciembre 2018]. Disponible en: [https://www.academia.edu/33038784/DISE%C3%91O\\_Y\\_CONSTRUCCI%C3%93N\\_DE\\_UN\\_SISTEMA\\_PARA\\_LA\\_DETECCI%C3%93N\\_DE\\_SE%C3%91ALES\\_ELECTROMIOGR%C3%81FICAS\\_TESIS\\_Presentada\\_por](https://www.academia.edu/33038784/DISE%C3%91O_Y_CONSTRUCCI%C3%93N_DE_UN_SISTEMA_PARA_LA_DETECCI%C3%93N_DE_SE%C3%91ALES_ELECTROMIOGR%C3%81FICAS_TESIS_Presentada_por)

**CONSEJO NACIONAL PARA LA IGUALDAD DE DISCAPACIDADES, ECUADOR.** Estadísticas de Discapacidad. [en línea]. 2018. [Consulta: 13 diciembre 2018]. Disponible en: <https://www.consejodiscapacidades.gob.ec/estadisticas-de-discapacidad/>.

**CONSEJO NACIONAL DE FOMENTO EDUCATIVO, MÉXICO.** Discapacidad auditiva Guía didáctica para la inclusión en educación inicial y básica. [en línea], 2010., pp. 71. Disponible en: <http://www.conafe.gob.mx/educacioncomunitaria/programainclusioneducativa/discapacidad-auditiva.pdf>.

**DE LUCA, C., ADAM, A., WOTIZ, R., DONALD, L., HAMID, S. y LUCA, D.**, "Decomposition of Surface EMG Signals". *J Neurophysiol* [en línea], 2006, vol. 96, pp. 1646-1657. [Consulta: 9 octubre 2018]. DOI 10.1152/jn.00009.2006. Disponible en: [www.jn.org](http://www.jn.org).

**GAO, R. y YAN, R.**, "Non-stationary signal processing for bearing health monitoring". *International Journal of Manufacturing Research*, [en línea] 2006, vol. 1, no. 1, pp. 18-40. [Consulta: 9 octubre 2018]. ISSN 17500605. DOI 10.1504/IJMR.2006.010701. Disponible en: [https://www.researchgate.net/publication/220650705\\_Non-stationary\\_signal\\_processing\\_for\\_bearing\\_health\\_monitoring](https://www.researchgate.net/publication/220650705_Non-stationary_signal_processing_for_bearing_health_monitoring).

**GUZMÁN, D.**, Guante Electrónico para traducir de lenguaje de señas a caracteres con voz artificial y conexión inalámbrica a dispositivos móviles para personas con discapacidad auditiva y de lenguaje en la Universidad Técnica de Ambato. (Tesis), (Pregrado), [en línea], Universidad Técnica de Ambato, Ambato- Ecuador, 2017. [Consulta: 9 octubre 2018]. Disponible en: <https://www.mendeley.com/catalogue/guante-electronico-para-traducir-lenguaje-señas-caracteres-con-voz-artificial-y-conexion-inalambrica/>

**HERMENS, H.J., FRERIKS, B., MERLETTI, R., STEGEMAN, D., BLOK, J., RAU, G., DISSELHORST-KLUG, C. y HÄGG, G.** "European Recommendations for Surface ElectroMyoGraphy Results of the SENIAM project". *Roessingh research and development* [en línea]. 1999. [Consulta: 13 diciembre 2018]. Disponible en: <https://pdfs.semanticscholar.org/1ab2/8b8afcb1216cab1b2f8da0de246c3d5ed6e8.pdf>.

**INFOTEC.** Sistemas Embebidos: Innovando hacia los Sistemas Inteligentes. [en línea]., 2016, [Consulta: 15 febrero 2019]. Disponible en: [http://www.semanticwebbuilder.org.mx/es\\_mx/swb/Sistemas\\_Embebidos\\_Innovando\\_hacia\\_los\\_Sistemas\\_Inteligentes\\_](http://www.semanticwebbuilder.org.mx/es_mx/swb/Sistemas_Embebidos_Innovando_hacia_los_Sistemas_Inteligentes_).

**KONRAD, P.** The ABC of EMG A Practical Introduction to Kinesiological Electromyography. [en línea]. 2006. Scottsdale, Arizona 85254: Disponible en: [www.konr.noraxon.com](http://www.konr.noraxon.com).

**KOSMIDOU, V. y HADJILEONTIADIS, L.** "Sign language recognition using intrinsic-mode sample entropy on sEMG and accelerometer data". *IEEE Transactions on Biomedical Engineering*, 2009, vol. 56, no. 12, pp. 2879-2890. ISSN 00189294. DOI 10.1109/TBME.2009.2013200.

**LEE, G., GOMMERS, R., WASELEWSKI, F., WOHLFAHRT, K. y LEARY, A.,** "PyWavelets: A Python package for wavelet analysis". *Journal of Open Source Software*, 2019, vol. 4, no. 36, pp. 1237. ISSN 2475-9066. DOI 10.21105/joss.01237.

**MARTÍNEZ, M., PÉREZ, M., PADILLA, D., LÓPEZ-LIRIA, R. y ACIÉN, F.L.** "Métodos De Intervención En Discapacidad Auditiva". *International Journal of Developmental and Educational Psychology* [en línea], 2008, vol. Volumen 3, no. 1, pp. 219-224. DOI 01050 [pii]. Disponible en: [http://infad.eu/RevistaINFAD/2008/n1/volumen3/INFAD\\_010320\\_219-224.pdf](http://infad.eu/RevistaINFAD/2008/n1/volumen3/INFAD_010320_219-224.pdf).

**MATICH, D.** Redes Neuronales: Conceptos Básicos y Aplicaciones. [en línea], 2001, pp. 55. Disponible en: <ftp://decsai.ugr.es/pub/usuarios/castro/Material-Redes-Neuronales/Libros/matich-redesneuronales.pdf>.

**MERLETTI, R. y PARKER, P.A.** *Electromyography: Physiology, Engineering, and Noninvasive Applications*. New Jersey- EEUU. 2004, ISBN 0-471-67580-6.

**MINISTERIO DE EDUCACIÓN CHILE.** Guía de apoyo técnico-pedagógico: Necesidades educativas especiales en el nivel de educación parvularia. *Primera Edición* [en línea], 2007, pp. 46. Disponible en: <http://portales.mineduc.cl/usuarios/edu.especial/File/GuiaAuditiva.pdf>.

**PLAN NACIONAL DEL BUEN VIVIR, ECUADOR** Objetivo 2 : Auspiciar la igualdad , la cohesión , la inclusión y la equidad social y territorial en la diversidad Objetivo 4 : Fortalecer las capacidades y potencialidades de la ciudadanía, 2017.

**OSUNA, S., GONZALEZ, J. y ILZARBE, A.** "Procesado de señales EMG en Trastornos Neuromusculares". *ResearchGate*, no. JUNE 2013, pp. 0-24. DOI 10.13140/2.1.4902.9445.

**OVIEDO, A., CARRERA, X. y CABEZAS, R..** Ecuador, atlas sordo – Cultura Sorda. [en línea], 2015, [Consulta: 18 septiembre 2018]. Disponible en: <http://www.cultura-sorda.org/ecuador-atlas-sordo/>.

**PÉREZ, M.** Modelado de descargas eléctricas en gases mediante redes neuronales. (Tesis),(Maestría), [en línea]. Universidad de Sevilla, Sevilla-España, 2010, [Consulta: 18 septiembre 2018]. Disponible en: <http://bibing.us.es/proyectos/abreproy/70196/>

**PÉREZ, P.** Reconocimiento de imágenes del lenguaje de señas mexicano. (Tesis),(Maestría), [en línea]. Instituto Politécnico Nacional, Distrito Federal-México, 2016, [Consulta: 18 septiembre 2018]. Disponible en: <http://148.204.63.111/SABERv3/Repositorios/webVerArchivo/26113>

**MARTÍNEZ, J.** Estudio comparativo de modelos de machine learning para la detección de dianas microARN. (Tesis),(Maestría), [en línea]. Universitat Oberta de Catalunya, España, 2018, [Consulta: 18 septiembre 2018]. Disponible en: <https://pdfs.semanticscholar.org/a5ff/f61a20c2f6d1e489f15138e875003f40d854.pdf>

**POLOLU CORPORATION.** MyoWare Muscle Sensor. [en línea]. 2014, [Consulta: 14 octubre 2018]. Disponible en: <https://www.pololu.com/product/2732>.

**REGISTRO OFICIAL ÒRGANICO GENERAL DEL ECUADOR..** Suplemento al Registro Oficial No. 796 - Registro Oficial del Ecuador 2012.

**TOO, J., ABDULLAH, A.R., MOHD SAAD, N., MOHD ALI, N. y MUSA, H.** "A detail study of wavelet families for EMG pattern recognition". *International Journal of Electrical and Computer Engineering*, 2018, vol. 8, no. 6, pp. 4221-4229. ISSN 20888708. DOI 10.11591/ijece.v8i6.pp.4221-4229.

**VISCONTI, P., GAETANI, F., ZAPPATORE, G. y PRIMICERI, P.** "Technical features and functionalities of Myo armband: An overview on related literature and advanced applications of

myoelectric armbands mainly focused on arm prostheses". *International Journal on Smart Sensing and Intelligent Systems*, 2018, vol. 11, no. 1, pp. 1-25. ISSN 11785608. DOI 10.21307/ijssis-2018-005.

**ZAMORANO CASTAÑO, M.** Analisis De Señales Mediante STFT Y Wavelet . Aplicación a defectología en rodamientos. [en línea], (Tesis), (Pregrado), Universidad Caarlos III de Madrid. Madrid-España, 2010. [Consulta: 14 octubre 2018]. Disponible en: <https://e-archivo.uc3m.es/handle/10016/10849>

## ANEXOS:

### Anexo A: Error cuadrático medio

Nº	A	B	C	D	E	F	G	H	I	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
1	0.00	0.12	0.57	0.96	0.00	0.17	0.00	4.08	0.05	0.00	0.44	0.12	0.00	0.29	0.00	0.00	0.00	0.32	0.03	0.44	0.53	0.12	0.00	0.01
2	0.02	0.17	0.05	0.67	0.40	0.04	0.05	0.29	0.06	3.71	0.53	0.00	0.10	0.03	0.06	0.29	0.08	0.08	0.40	0.02	0.04	0.04	0.00	0.00
3	0.00	0.48	0.06	0.04	0.00	0.67	0.26	1.29	0.05	0.36	0.32	0.06	0.03	0.05	0.06	0.00	0.00	0.22	0.06	0.01	0.84	0.06	0.08	0.04
4	0.00	0.29	0.03	0.02	0.44	0.67	0.00	0.08	0.14	0.00	0.08	0.08	0.03	0.06	0.32	0.10	0.05	0.01	0.04	0.00	0.10	0.73	0.04	0.01
5	0.04	0.05	0.04	0.06	0.12	0.96	0.00	0.00	0.17	0.32	0.17	0.40	0.01	0.00	0.00	0.03	0.08	0.17	0.73	0.06	0.08	0.00	0.17	0.00
6	0.06	0.17	0.04	0.00	0.78	0.05	0.05	0.04	0.53	1.00	0.04	0.04	4.48	0.01	0.14	0.00	0.17	0.05	0.10	1.09	0.22	0.78	0.04	0.06
7	0.12	0.06	0.01	0.05	2.60	0.14	0.01	0.10	0.17	0.67	0.17	0.14	0.04	0.22	0.17	0.05	0.03	0.00	0.36	0.00	0.22	0.84	0.01	0.04
8	0.00	1.37	0.01	0.00	0.29	0.00	0.02	0.01	0.01	0.04	0.00	0.06	0.17	1.93	0.00	0.06	0.12	0.05	0.02	0.04	0.01	1.29	2.11	0.05
9	0.05	2.20	0.10	0.05	0.57	0.53	0.05	0.08	0.00	0.53	0.00	0.02	0.00	0.17	0.17	0.14	0.05	0.04	0.78	0.17	0.01	0.01	0.06	0.04
10	0.57	1.02	0.06	0.04	2.60	0.14	0.01	0.84	0.22	1.76	0.02	0.05	1.22	0.00	0.36	0.03	0.04	0.14	0.05	0.04	0.04	0.06	0.00	0.12
11	0.00	0.03	0.00	0.05	5.03	0.14	0.00	0.62	1.02	0.67	1.29	0.14	0.04	0.20	0.12	0.03	0.03	1.29	0.12	0.14	0.14	0.05	0.00	0.20
12	0.00	0.20	0.03	0.01	0.96	0.00	0.00	2.40	0.20	0.00	0.00	0.00	0.06	0.04	0.08	0.00	0.00	0.06	3.02	0.32	0.10	0.06	0.02	0.04
13	0.00	5.47	0.05	0.12	0.26	0.08	0.05	3.02	0.22	1.85	0.26	0.05	0.00	0.14	0.00	0.10	0.02	0.53	0.02	0.03	0.04	0.00	1.15	0.48
14	0.04	0.05	0.53	0.06	1.22	0.17	0.03	1.02	0.44	0.14	0.53	0.04	0.10	0.10	0.67	0.01	0.02	0.02	0.04	2.60	0.02	0.62	1.02	0.01

15	0.01	0.36	0.00	0.04	0.01	1.15	0.00	0.29	0.06	0.02	0.00	0.17	0.02	0.06	0.03	0.04	0.32	0.62	0.02	0.57	1.09	0.04	0.00	0.01
16	0.00	1.02	0.10	0.22	0.00	0.44	0.06	0.73	1.68	0.84	0.01	0.10	0.73	0.62	0.00	0.00	0.02	0.00	0.04	0.12	0.40	0.20	0.12	0.10
17	0.01	2.91	0.22	0.02	0.32	0.06	0.00	0.90	0.22	0.36	0.04	0.02	0.12	0.14	0.12	0.02	0.01	0.20	0.29	1.29	0.53	0.14	0.00	0.00
18	0.00	6.23	0.00	0.06	0.12	2.02	0.00	2.20	0.73	0.03	0.05	0.06	0.01	0.00	0.10	0.48	0.03	0.02	0.57	1.76	0.14	0.14	3.02	0.06
19	0.03	0.29	0.01	0.00	4.09	0.78	0.00	0.67	0.20	0.73	0.01	0.01	0.14	0.40	0.04	0.00	0.02	0.08	0.44	1.09	0.62	0.04	0.03	0.01
20	0.00	0.40	0.03	0.12	0.01	0.67	0.00	0.78	0.01	1.85	0.03	0.06	0.02	0.05	0.00	0.00	0.00	0.05	0.14	0.06	0.17	0.01	0.00	0.00



**Anexo B:** Número de datos verdaderos por cada clase vs número de datos clasificados correctamente por cada modelo

Nombre Real	Salida	N° de letras por cada clase	N° de letras clasificadas correctamente		
			MLP	DT	RF
A	1	<b>18</b>	18	17	18
B	2	<b>17</b>	13	13	17
C	3	<b>12</b>	9	9	11
D	4	<b>13</b>	8	7	10
E	5	<b>16</b>	15	13	15
F	6	<b>16</b>	14	12	15
G	7	<b>14</b>	13	13	14
H	8	<b>15</b>	15	15	15
I	9	<b>11</b>	11	10	11
K	11	<b>18</b>	18	18	18
L	12	<b>17</b>	16	10	16
M	13	<b>7</b>	7	6	7
N	14	<b>17</b>	16	15	17
O	15	<b>13</b>	13	12	12
P	16	<b>10</b>	10	10	10
Q	17	<b>14</b>	14	10	14
R	18	<b>15</b>	15	14	15
S	19	<b>17</b>	16	17	16
T	20	<b>17</b>	14	10	15
U	21	<b>7</b>	4	4	5
V	22	<b>9</b>	9	6	6
W	23	<b>16</b>	14	12	14
X	24	<b>12</b>	10	9	11
Y	25	<b>15</b>	14	14	15

**Anexo C:** Métrica de evaluación: precisión y recall de cada modelo

		MLP		Árbol de decisión		Random Forest	
Nombre Real	Salida	precisión	recall	precisión	recall	precisión	Recall
A	1	1	0.95	0.94	0.89	1	0.95
B	2	0.76	0.76	0.76	0.76	1	0.89
C	3	0.75	0.75	0.75	0.64	0.92	0.92
D	4	0.62	0.73	0.54	0.88	0.77	1
E	5	0.94	0.94	0.81	0.65	0.94	0.88
F	6	0.88	0.93	0.75	1	0.94	0.94
G	7	0.93	1	0.93	0.81	1	1
H	8	1	0.94	1	0.94	1	1
I	9	1	1	0.91	1	1	0.92
K	11	1	1	1	0.9	1	1
L	12	0.94	1	0.59	0.83	0.94	1
M	13	1	1	0.86	0.75	1	1
N	14	0.94	0.94	0.88	0.83	1	1
O	15	1	0.87	0.92	0.86	0.92	0.86
P	16	1	1	1	1	1	1
Q	17	1	1	0.71	0.91	1	0.93
R	18	1	1	0.93	0.88	1	0.94
S	19	0.94	0.89	1	0.94	0.94	1
T	20	0.82	0.78	0.59	0.67	0.88	0.94
U	21	0.57	1	0.57	1	0.71	0.62
V	22	1	0.69	0.67	0.67	0.67	0.75
W	23	0.88	0.93	0.75	0.8	0.88	0.93
X	24	0.83	0.83	0.75	0.64	0.92	1
Y	25	0.93	1	0.93	0.88	1	1
PROMEDIO		0.90541667	0.91375	0.81416667	0.83875	0.93458333	0.93625

**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**DIRECCIÓN DE BIBLIOTECAS Y RECURSOS PARA EL APRENDIZAJE**  
**Y LA INVESTIGACIÓN**  
**UNIDAD DE PROCESOS TÉCNICOS**  
**REVISIÓN DE NORMAS TÉCNICAS, RESUMEN Y BIBLIOGRAFÍA**

**Fecha de entrega:** 31 / Octubre / 2019

<b>INFORMACIÓN DEL AUTOR/A (S)</b>
<b>Nombres – Apellidos:</b> Ana María Ñacato Estrella
<b>INFORMACIÓN INSTITUCIONAL</b>
<b>Facultad:</b> Informática y Electrónica
<b>Carrera:</b> Ingeniería Electrónica en Control y Redes Industriales
<b>Título a optar:</b> Ingeniera en Electrónica en Control y Redes Industriales
<b>f. Documentalista responsable:</b>